



Escola Politècnica Superior  
d'Enginyeria de Manresa

UNIVERSITAT POLITÈCNICA DE CATALUNYA

# **TREBALL FINAL DE GRAU**

Monitor d'energia elèctrica, estudi de  
viabilitat i prototip de baix cost

**Nom** : Luis Ernesto Chaparro Peña

**Especialitat** : Grau en enginyeria Electrónica Industrial i Automàtica

**Tutor** : Víctor Barcons Xixons

**Convocatòria** : Juliol 2015

# Índex

Índex de imatges .....	3
1 Agraïments .....	7
2 Resum .....	8
3 Introducció .....	10
3.1 Elecció del tema del Treball Final de Grau .....	10
3.2 Objectius del TFG .....	10
3.3 Què és un monitor d'energia elèctrica? .....	11
3.4 Com funciona el prototip del monitor d'energia elèctrica? .....	12
3.5 Quina és la seva utilitat? .....	14
3.6 Diferències entre un monitor d'energia elèctrica comercial i el prototip del monitor d'energia d'aquest projecte .....	15
4 Parts del monitor d'energia elèctrica .....	16
4.1 Placa ToolStick F330 DC .....	16
4.2 Sensor de corrent .....	19
4.3 Comunicació entre el microcontrolador i el PC .....	21
4.3.1 Nivells lògics de tensió .....	21
4.3.2 Comunicació amb fils .....	22
4.3.3 Comunicació sense fils .....	24
4.4 Circuit analògic .....	26
4.4.1 Disseny del circuit analògic .....	26
4.4.1.1 Disseny adaptació del senyal de tensió d'entrada .....	26
4.4.1.2 Disseny del circuit per alimentar el sensor de corrent .....	31
4.4.2 Disseny de la placa de circuit imprès (PCB) .....	32

5 Programació .....	37
5.1 Programació del microcontrolador .....	37
5.1.1 Explicació del codi .....	37
5.1.2 Simulació del funcionament del codi amb el software µVision IDE .....	60
5.1.3 Comprovació del funcionament del codi a la placa ToolStick F330 DC amb el software Silicon Labs IDE .....	63
5.2 Programació de l'adquisició de dades amb el software LabVIEW .....	65
5.2.1 Explicació del codi .....	65
5.2.2 Comprovació del funcionament del codi .....	69
6 Muntatge final .....	70
6.1 Consideracions durant el muntatge final .....	69
6.2 Realització de proves .....	75
7 Pressupost .....	79
8 Estudi de viabilitat .....	81
8.1 A quins clients aniria destinat el producte? .....	81
8.2 Com es donaria a conèixer el producte? .....	81
8.3 Principals competidors .....	81
8.4 Es viable el producte? .....	82
9 Conclusions .....	83
10 Bibliografia .....	85

## Índex de imatges

Il·lustració 1 – Monitor d'energia elèctrica .....	10
Il·lustració 2 – Suma de Riemann en una funció sinusoidal .....	12
Il·lustració 3 – Suma de Riemann en una funció constant .....	12
Il·lustració 4 – Regleta per reduir els consums Stand by .....	13
Il·lustració 5 – Placa ToolStick F330 DC .....	15
Il·lustració 6 – Circuit on hi incorporat el microcontrolador .....	16
Il·lustració 7 – Eina des de la qual l'usuari rebrà o enviarà informació .....	16
Il·lustració 8 – Esquema funcional del convertidor Analògic-Digital .....	17
Il·lustració 9 – Efecte Hall .....	18
Il·lustració 10 – Sensor de corrent .....	18
Il·lustració 11 – Datasheet del sensor de corrent .....	19
Il·lustració 12 – Taula amb els diferents nivells lògics .....	20
Il·lustració 13 – Distribució de pins del port sèrie del PC .....	21
Il·lustració 14 – Diagrama de blocs 8051F330 .....	22
Il·lustració 15 – Esquema de connexió microcontrolador, PC i MAX232 .....	22
Il·lustració 16 – Mòdul Bluetooth mestre/esclau .....	23
Il·lustració 17 – Esquema de connexió sense fils entre microcontrolador, PC i MAX232 .....	23
Il·lustració 18 – Distribució de pines de l'optoacoblador TLP621 .....	25
Il·lustració 19 – Relació entre voltatge (VF) i corrent (IF) que se li pot aplicar al LED .....	25
Il·lustració 20 – Relació entre corrent (IF) que se li pot aplicar al LED i corrent al col·lector (IC) .....	25
Il·lustració 21 – Relació entre corrent al col·lector (IC) i tensió a la sortida del optoacoblador (VCE) .....	26

Il·lustració 22 – Forma del senyal que s’ha d’introduir al opto, quan es treballi amb un senyal d’entrada de 400 volts d’amplitud .....	26
Il·lustració 23 – Esquema i formula d’un divisor de tensió.....	26
Il·lustració 24 – Esquema i senyal del sortida del divisor de tensió fet amb el programa PSIM .....	27
Il·lustració 25 – Esquema i formula d’un amplificador operacional sumador no inversor .....	27
Il·lustració 26 – Esquema i senyal del sortida del amplificador sumador no inversor fet amb PSIM .....	27
Il·lustració 27 – Esquema i formula d’un amplificador operacional no inversor .....	28
Il·lustració 28 – Esquema i senyal del sortida del amplificador no inversor fet amb PSIM .....	28
Il·lustració 29 – Esquema i senyal del sortida del amplificador sumador no inversor fet amb PSIM .....	28
Il·lustració 30 – Taula de condicions d’operació recomanades .....	29
Il·lustració 31 – Esquema del circuit per adaptar el senyal d’entrada a la placa ToolStick F330 DC fet amb el programa EAGLE PCB .....	29
Il·lustració 32 – Esquema del circuit per alimentar el sensor de corrent fet amb PSIM .....	30
Il·lustració 33 – Esquema del circuit per alimentar el sensor de corrent fet amb EAGLE PCB .....	30
Il·lustració 34 – Circuit analògic sencer muntat sobre la placa protoboard .....	31
Il·lustració 35 – Esquema elèctric complet del circuit analògic .....	32
Il·lustració 36 – Realització del disseny de la PCB amb el programa EAGLE .....	32
Il·lustració 37 – Resultat del disseny de la PCB del monitor d’energia elèctrica .....	33
Il·lustració 38 – Ploter fresador que serveix per la fabricació de PCB’s .....	33
Il·lustració 39 – PCB sense components .....	34

Il·lustració 40 – PCB amb tots els components .....	34
Il·lustració 41 – Codi ASCII .....	41
Il·lustració 42 – Missatge del programa µVision per indicar que el codi no presenta problemes .....	59
Il·lustració 43 – Definició dels senyals virtuals al arxiu debug.ini .....	60
Il·lustració 44 – Definició dels senyals virtuals al arxiu debug.ini.....	61
Il·lustració 45 – Missatge del programa Silicon Labs IDE per indicar que el codi no presenta problemes .....	62
Il·lustració 46 – Comprovació del codi a la realitat utilitzant una aspiradora de 1,4 KW .....	63
Il·lustració 47 – Codi d'adquisició de dades complet .....	66
Il·lustració 48 – Fragment 1 del codi .....	66
Il·lustració 49 – Fragment 2 del codi .....	66
Il·lustració 50 – Fragment 3 del codi .....	67
Il·lustració 51 – Panell que adquireix les dades que li envia el microcontrolador .....	67
Il·lustració 52 – Adquisició de dades durant els primers 35 segons .....	68
Il·lustració 53 – Adquisició de dades durant els primers 51 segons .....	68
Il·lustració 54 – Blackout .....	69
Il·lustració 55 – Microtalls en el voltatge .....	69
Il·lustració 56 – Sag .....	69
Il·lustració 57 – Brownouts .....	70
Il·lustració 58 – Swell .....	70
Il·lustració 59 – Over voltage .....	70
Il·lustració 60 – Transients .....	70
Il·lustració 61 – Notching .....	71

Il·lustració 62 – Soroll elèctric .....	71
Il·lustració 63 – Harmònics .....	71
Il·lustració 64 – Canvi de la freqüència del senyal .....	71
Il·lustració 65 – Senyal de sortida del optoacobrador .....	72
Il·lustració 66 – Adaptador per eliminar la presa a terra .....	72
Il·lustració 67 – Sentit del corrent si no es posen els adaptadors .....	73
Il·lustració 68 – Sentit del corrent si es posen els adaptadors .....	73
Il·lustració 69 – Consum aspiradora durant els primers 50 segons .....	74
Il·lustració 70 – Consum aspiradora des de els 22 fins als 72 segons .....	74
Il·lustració 71 – Consum paellera elèctrica durant els primers 30 segons .....	75
Il·lustració 72 – Consum paellera elèctrica des de els 12 fins als 62 segons .....	75
Il·lustració 73 – Consum assecador de cabells durant els primers 45 segons .....	76
Il·lustració 74 – Consum assecador de cabells des de els 21 fins als 71 segons ....	76
Il·lustració 75 – Consum assecador de cabells des de 10 fins 25 segons i aspiradora des de 27 fins a 50 segons .....	77
Il·lustració 76 – Consum aspiradora des de 27 fins a 50 segons i conjunt aspiradora-assecador de cabells des de 60 fins a 76 segons .....	77

# 1 Agraïments

En aquest apartat vull donar les gràcies a totes aquelles persones que d'una manera o altre han contribuït a la realització d'aquest projecte.

Agrair al meu tutor, Victor Barcons Xixons, professor del Departament. Eng. Disseny i Programació de Sist. Electrònics, tant pel suport i l'ajuda proporcionada, així com també pel material necessari atorgat per al desenvolupament d'aquest projecte.

Agrair també al Manel Blanes Priego, encarregat del Departament. Eng. Disseny i Programació de Sist. Electrònics, per proporcionar-me els components necessaris per la construcció del circuit analògic, així com també per haver fabricat la placa de circuit imprès i per haver soldat els components a dita placa.

Finalment, agrair als meus familiars i amics tot el suport i comprensió obtingut durant la realització d'aquest projecte de final de grau.



## 2 Resum

Aquest projecte es basa en la realització d'un prototip d'un monitor d'energia elèctrica, que s'encarregarà de mesurar el consum d'energia d'aparells elèctrics, tant de corrent altern com de corrent continu.

La idea d'aquest monitor d'energia elèctrica es basa en l'obtenció de la potència dels aparells que es volen mesurar, per fer-ho s'agafa el senyal de tensió i el senyal de corrent que alimenta l'aparell, posteriorment s'agafen els valors analògics d'aquestes dos senyals i es multipliquen ( $\text{Potència elèctrica} = \text{Tensió} \times \text{Corrent}$ ), per finalment obtenir el valor de potència que està consumint l'aparell. A partir d'aquí es realitza el càlcul de l'energia elèctrica, que consisteix en multiplicar la potència que s'està consumint per el temps en el que s'està fent la mesura del consum elèctric ( $\text{Energia elèctrica} = \text{Potència elèctrica} \times \text{acumulació de temps}$ ).

Per construir el monitor d'energia elèctrica s'utilitzarà un sensor de corrent, que pot detectar tant corrent altern com corrent continu. A més s'utilitza la placa MCU ToolStick F330 DC, la qual incorpora un microcontrolador (circuit integrat que es pot reprogramar per controlar petits processos). En aquest cas es tracta del microcontrolador Intel 8051F330. La placa també incorpora un adaptador que permet la comunicació entre el microcontrolador i el PC a través d'una connexió USB. El fabricant de dita placa també ens proporciona una eina software que ens permet programar el microcontrolador tantes vegades com es vulgui, també permet veure els valors en temps reals de totes les variables que s'hagin creat, així com també permet emular un display i un teclat, que es necessita perquè l'usuari pugui introduir informació.

A més dels elements abans enunciats, es realitza el disseny d'un circuit analògic, el qual ens permet adaptar els senyals analògics per poder tractar-los amb la placa ToolStick F330 DC i així poder realitzar els càlculs de consum energètic correctament. Dit circuit primer es realitza a sobre d'una placa protoboard amb la que s'implementen les connexions del circuit a través de cables. Posteriorment perquè el circuit es vegi de manera més ordenada i més professional, el circuit s'implementa sobre una targeta de circuit imprès (PCB), la qual requereix un disseny previ. Per fer aquest disseny s'utilitza un software de llicència lliure anomenat CadSoft EAGLE PCB.

Per poder visualitzar els càlculs al PC, primer es necessita establir una comunicació per poder enviar la informació que genera el microcontrolador cap al PC. Aquesta comunicació és la sèrie RS232. A part d'aquesta comunicació es necessita un software que ens permeti crear una interfície amb la que es pugui visualitzar en temps real els valors i les gràfiques de la demanda actual de potència elèctrica i de la energia elèctrica que s'està consumint, així com també visualitzar el preu que està costant l'energia elèctrica i si la demanda de potència elèctrica excedeix un valor de potència elèctrica que estableix l'usuari. Per fer dita tasca, el software que s'utilitza és LabVIEW System Desing.

Una vegada s'ha realitzat tot el muntatge físic del monitor d'energia elèctrica, s'enuncia de manera teòrica com es pot establir una comunicació sense fils entre microcontrolador i el PC.

## Summary

This project is based on the realization of a prototype of a monitor of electrical energy, which is responsible for measuring the energy consumption of electrical appliances, both alternating current and direct current.

The idea of this electrical energy monitor is based on obtaining the power of the appliances that you want to measure, to do this you take the signal voltage and current signal that powers the device, then we take the analogue values of these two signals and it are multiplying ( $\text{Electrical Power} = \text{Voltage} \times \text{Current}$ ), to finally get the value of power that is consuming the device. From here it performs the calculation of the electrical energy, which is to multiply the power that is consumed by the time in which it is carrying out the measurement of the power consumption ( $\text{Electric Power} = \text{power} \cdot \text{time accumulation}$ ).

To build the power monitor will be used a current sensor that can detect both alternating current and direct current. In addition we will use the MCU Board ToolStick F330 DC, which incorporate a microcontroller (integrated circuit that can be reprogrammed to control small processes). In this case it is the Intel 8051F330 microcontroller. The Board also includes an adapter that allows communication between the microcontroller and the PC through a USB connection. The manufacturer of this Board also provides a software tool that allows us to program the microcontroller as many times as you want to, also it allows you to view real time values of all the variables that have been created, as well as allows you to emulate a display and a keyboard, which is needed so that the user can enter information.

In addition of the before set forth elements, it make a design of analogue circuit to adapt the analogue signals and of this way we can treat them with the plate ToolStick F330 DC and after, do the calculations of energetic consumption properly. Said circuit is makes on the protoboard board, where the connections will be done through cables. After, we will do the same circuit but this time in a printed circuit board, that previously needs a design. To do this design we use a software of free license called CadSoft EAGLE PCB.

In order to view the calculations in the PC, first we need to establish communication to send the information generated by the microcontroller to the PC. This is the serial RS232. Further of this communication you will need a software that allows us to create an interface with which we can view in real time the values and graphs of the current demand of electric power and also electrical energy that is consuming, as well as view the price that is costing the electrical energy and if the demand for electric power exceeds a value of electric power that sets the user. To do this task, the software that you use is LabVIEW System Design.

Once you have done all the physical mounting of the monitor of electric power, we will enunciate the theoretical way of how you can establish a wireless communication between microcontroller and PC.

### 3 Introducció

L'objectiu principal d'aquest apartat és explicar al lector els punts que defineixen i deixen clar quin és el tema d'aquest projecte. Així mateix es comencen a explicar una sèrie de preguntes que el lector es podria plantejar, una vegada sap quin és el tema del projecte.

#### 3.1 Elecció del tema del Treball Final de Grau

Quan va arribar el moment d'escollir quin seria el tema del Treball Final de Grau, sincerament no sabia quin tema escollir degut a l'ampli rang d'aplicacions de l'electrònica. A més a més havia de ser un tema que estigués al meu abast, que no comportés una despesa de diners elevat i a més que no fos un tema que molta gent escollís. Degut a que no hem decidia vaig decidir demanar consell al meu tutor Victor Barcons.

En un primer moment li vaig dir que m'agradaria fer qualsevol cosa que tingués a veure amb la programació de microcontroladors, ja que era un aspecte que m'agradava bastant i a més perquè era un aspecte que es va treballar en 2 assignatures ( Microcomputadors i Programació de baix nivell: aplicacions industrials dels microcontroladors ). Així doncs, volia aprofitar els coneixements que vaig adquirir les assignatures dites i el meu interès sobre aquest aspecte.

Després d'exposar-li les meves preferències sobre el tema del projecte, el tutor em va proposar de fer un monitor d'energia elèctrica. Em va semblar un tema interessant, entretingut, que s'adaptava a les meves preferències i que no era un tema que la gent escollia.

#### 3.2 Objectius del TFG

Una vegada vaig decidir quin seria el tema del Treball Final de Grau, em vaig fixar una sèrie d'objectius que m'ajudessin a confeccionar un projecte sòlid. Aquests objectius són:

- Tenir clar tots conceptes relacionats amb el funcionament d'un monitor d'energia elèctrica.
- Saber dividir les parts que constitueixen un monitor d'energia elèctrica.
- Reforçar els coneixements de programació adquirits en les assignatures.
- Realització d'un prototip d'un monitor d'energia elèctrica de baix cost.
- Assegurar un bon funcionament del prototip dins d'uns límits acceptables.

### 3.3 Què és un monitor d'energia elèctrica?

L'electricitat és molt important per la humanitat a dia d'avui, tant és així que qualsevol aparell funciona o gran part del seu funcionament és gràcies a aquest fenomen físic. Degut a la seva importància, la indústria elèctrica exigeix als consumidors un preu considerable, i conforme van passant els anys aquest preu és ascendent. Aquest factor és negatiu per als consumidors, però si ho pensem bé hi ha un altre factor negatiu que aquest si ens afecta a tots els éssers vius, ens referim al medi ambient.

Al llarg dels anys i degut a la demanda ascendent d'energia elèctrica l'home ha trobat la manera de generar-la a través de diferents maneres, però que per desgracia d'una manera o altre afecten al medi ambient, a continuació s'enuncien les diferents maneres més comuns d'avui dia per generar energia elèctrica així com també com afecten al medi ambient.

**Combustió de fòssils per la generació d'energia elèctrica:** L'ús de combustibles fòssils genera emissions de gasos d'efecte hivernacle i de pluja àcida a l'atmosfera, al costat de partícules volants que poden contenir metalls pesats. Afecten negativament als ecosistemes fluvials a causa dels abocaments d'aigua calenta en aquests.

**Aprofitament dels salts d'aigua generació d'energia elèctrica:** Grans centrals hidroelèctriques que generen dipòsits realment poden produir enormes quantitats de metà i CO<sub>2</sub> a l'atmosfera. Els embassaments creats per les preses grans i mega repeses destrueixen els hàbitats locals.

**Generació d'energia elèctrica a través de centrals nuclears:** El principal inconvenient és la generació de residus tòxics aquests poden romandre actius durant centenars de milers d'anys. Actualment, gran part de les deixalles radioactives de les plantes ha estat guardades a la planta mateixa.

Degut a aquests factors negatius surt la necessitat de consumir una quantitat justa i necessària d'energia. És aquí on intervenen els monitors d'energia elèctrica, els quals ens poden ajudar a reduir el nostre consum pel bé de la nostra butxaca i pel medi ambient. Així doncs un monitor d'energia elèctrica és un aparell que serveix per mesurar el consum elèctric d'un aparell o d'una instal·lació elèctrica d'un habitatge.



Il·lustració 1 – Monitor d'energia elèctrica

### 3.4 Com funciona el prototip del monitor d'energia elèctrica?

El monitor d'energia elèctrica que es realitza en aquest projecte, es basa en el càlcul del valor eficaç del senyal de tensió d'entrada i del valor eficaç del senyal del sensor de corrent. Així doncs, el primer s'ha de saber és com es calcula el valor eficaç de tensió de qualsevol senyal, el qual es calcula a través de la següent fórmula:

$$V_{ef} = \sqrt{\frac{1}{T} \int_{t_0}^{t_0+T} v^2(t) dt}$$

On:

$V_{ef}$  = Valor eficaç

$T$  = Període del senyal

$t_0$  = Temps inicial

$v^2(t) dt$  = Valor de tensió al quadrat en funció del temps

Per calcular la integral s'utilitzarà la suma de Riemann, que consisteix en traçar un nombre finit de rectangles dins d'un àrea irregular, calcular l'àrea d'aquests i finalment sumar-los. Com més gran sigui el nombre de rectangles menys error hi haurà en el càlcul de l'integral. En aquest cas el nombre de rectangles que s'ha definit pel càlcul de la integral és de 200. Per definir la base dels rectangles, primer s'ha d'establir la freqüència de amb la que es vol treballar majoritàriament, en aquest es treballarà a una freqüència de 50 Hz, ja que és la freqüència del senyal de corrent altern que et proporciona la companyia elèctrica per a les instal·lacions domèstiques. Després es calcula la freqüència de mostreig per saber quantes mostres s'està agafant per segon.

#### Període del senyal

$$T = \frac{1}{f} = \frac{1}{50} = 0,02 \text{ s}$$

#### Base dels rectangles

$$\Delta t = \frac{T}{\text{Rectangles}} = \frac{0,02}{200} = 0,1 \text{ ms}$$

#### Freqüència de mostreig

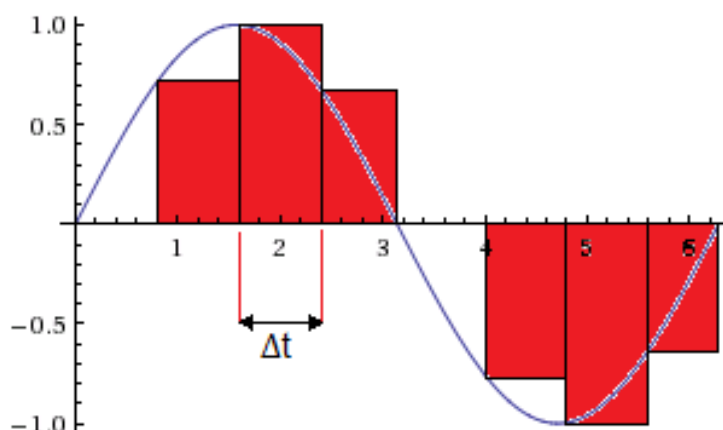
$$f = \frac{1}{\text{base rectangles}} = \frac{1}{0,1 \text{ ms}} = 10 \text{ Khz} = 10.000 \text{ mostres/s}$$

Ara es necessita saber les alçades dels rectangles, les quals estan condicionades per la forma de la senyal. Per saber dita alçada s'ha de tenir en compte els diferents mètodes que hi he en la suma de Riemann, les quals són:

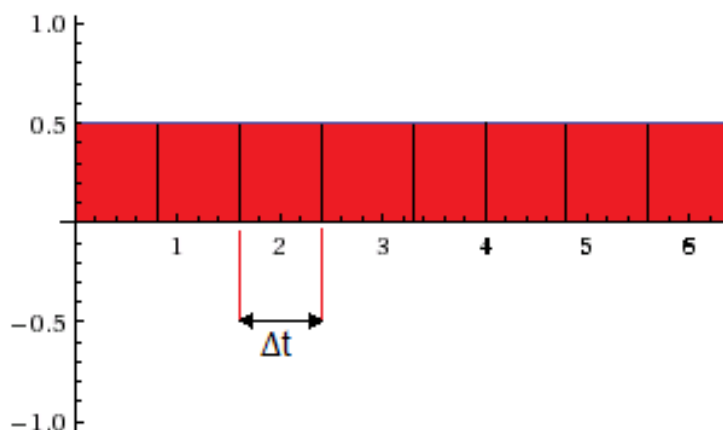
- 1- Agafar el punt inicial de l'interval de temps i veure el valor analògic que li correspon.
- 2- Agafar el punt final de l'interval de temps i veure el valor analògic que li correspon.
- 3- Agafar el valor mig entre el valor inicial de temps i el valor final i veure el valor analògic que li correspon.

En aquest cas s'agafa el mètode numero 1, agafar el valor inicial de l'interval de temps i veure en la funció quin valor analògic li correspon.

Amb la suma de Riemann es pot calcular l'àrea de qualsevol funció continua, el que implica que es pugui calcula el valor eficaç tant de funcions de corrent altern com funcions de corrent continu. A continuació es veurà un exemple de la suma de Riemann amb una funció sinusoidal (AC) i amb una funció constant (DC), utilitzant 8 rectangles per període del senyal.



Il·lustració 2 – Suma de Riemann en una funció sinusoidal



Il·lustració 3 – Suma de Riemann en una funció constant

Una vegada es té el càlcul de la integral i consegüentment el valor eficaç de tensió i del sensor de corrent, es multiplicant aquest dos valors per obtenir la demanda de potència actual. Per calcular la energia acumulada en de tenir en compta la seva formula:

$$E = P \cdot t$$

On:  
E = Energia elèctrica  
P = Potència elèctrica  
t = Temps

Com que l'energia va en funció del temps que s'ha fet és acumular i actualitzar el valor de la demanda de potència actual cada període de senyal, és a dir, cada 0,02 segons i el valor de la energia s'acumula i s'actualitza cada segon. Pel que fa referent al preu s'acumula i s'actualitza cada segon com l'energia. Cal comentar que el resultat del preu de la energia consumida és la multiplicació de l'energia acumulada pel valor del cost que cobra la companyia elèctrica per cada KWh. Dit valor està al voltant de 0,13 €/KWh, tot i així en el monitor d'energia d'aquest projecte l'usuari podrà introduir el preu del KWh, ja que aquest és diferent en cada companyia elèctrica.

Per poder veure la trajectòria del consum, es realitzen dos gràfiques una per la demanda de potència actual i un altre per l'energia acumulada, d'aquesta manera es pot veure ens quins instants de temps s'obté el major i el mínim consum. També s'incorpora l'opció perquè l'usuari determini un valor màxim de demanda de potència, i en el cas que la demanda de potència actual sobrepassi aquest valor al panell s'encendrà un llum de color vermell. En el cas que el valor de estigui per sota del valor que ha determinat l'usuari s'encendrà un llum de color verd.

### 3.5 Quina és la seva utilitat?

Els monitors d'energia elèctrica proporcionen una sèrie d'avantatges que permeten al usuari estalviar diners i com a conseqüència reduir l'impacte indirecte sobre el medi ambient, dites avantatges són:

- Conèixer en temps real el consum energètic i el cost d'una instal·lació domestica.
- Permet modificar els hàbits de consum, el que implica a l'usuari un benefici econòmic, ja que aquest passa a consumir només l'energia que necessita.
- Conèixer la demanda màxima de la seva instal·lació, per saber quina potència contractada és més adient per a les seves necessitats de consum.
- Conèixer els horaris on l'usuari consumeix més energia i així d'aquesta manera pot escollir quina és la tarifa més adient per satisfer les seves necessitats de consum.
- Permet detectar els consums fantasmes que es deuen al consum dels aparells quan estan en mode Stand by, d'aquesta manera l'usuari pot conèixer el cost anual que l'implicaria en la seva factura i podria resoldre'l amb una regleta.



Il·lustració 4 – Regleta per reduir els consums Stand by



### 3.6 Diferències entre un monitor d'energia elèctrica comercial i el prototip del monitor d'energia d'aquest projecte

En aquest apartat s'enuncien les principals diferències entre un monitor d'energia comercial i el prototip del monitor d'energia d'aquest projecte. Aquestes diferències s'expliquen a continuació :

- En la majoria de monitors d'energia comercials es suposa sempre que la tensió d'entrada serà una de tipus altern amb un valor eficaç de 230 V i amb un valor màxim de 325 V, aquesta tensió és la que arriba sempre a totes les instal·lacions domèstiques. Això vol dir que aquests monitors estan destinats mesurar només consums energètics d'instal·lacions domèstiques. Pel que fa referent al prototip del monitor d'energia d'aquest projecte, la tensió d'entrada no es suposa sinó que es calcula, la manera de calcular aquest valor s'ha explicat en l'apartat anterior (a través de les sumes de Riemann). El rang de tensió d'entrada del prototip anirà de 0 a 400 V, ja sigui una tensió de corrent altern com de corrent continu. Això vol que com a màxim, pot treballar amb una tensió DC de 400 V o amb una tensió AC de valor màxim 400 V i valor eficaç de 283 V. Així doncs el prototip podria mesurar la producció d'energia de plaques fotovoltaïques (DC) així com mesurar el consum energètic d'una instal·lació domèstica (AC). Cal comentar que les proves del prototip no es faran sobre una instal·lació domèstica, sinó que es faran sobre una regleta, en la que es mesurar el consum energètic de dos aparells.
- Pel que fa referent a la mesura de la intensitat consumida, el monitor d'energia comercial utilitza una pinça amperimètrica (que és en definitiva un sensor de corrent), que calcula el pas de corrent de la fase de la instal·lació gràcies al efecte Hall ( fenomen que s'explicarà més endavant ). Aquesta intensitat depèn de la potència contractada que hagi escollit l'usuari de la instal·lació. Per exemple si es té una potència contractada de 14,5 KW (la màxima per un habitatge) , per una instal·lació monofàsica, la màxima intensitat que podrà consumir la instal·lació serà de:

$$\text{Potència} = \text{Voltatge} \cdot \text{Intensitat} \quad \longrightarrow \quad \text{Intensitat} = \frac{\text{Potència}}{\text{Voltatge}} = \frac{5500}{230} = 63 \text{ A}$$

Això significa que els monitors d'energia comercials mesuraran com a màxim una intensitat de 63 A (valor eficaç). Una vegada té el valor de la intensitat l'envia a una pantalla a través d'una comunicació sense fils, en la majoria utilitzant connexió Wi-Fi. Quan el valor de intensitat arriba a la pantalla es multiplica per la tensió d'entrada, que sempre és un valor fix, 230 V i així obtenir la demanda de potència i consegüentment la energia consumida i el preu que està costant aquesta energia. Pel que fa referent al prototip d'aquest projecte, en lloc d'utilitzar una pinça amperimètrica, s'utilitza un sensor d'efecte Hall que podrà mesurar com a màxim una intensitat de 100 A (100 A en DC i 100 A de valor màxim en AC). Així doncs, el prototip podria calcular el consum de qualsevol instal·lació monofàsica d'un habitatge com la del exemple anterior (63 A de consum) i podria també mesurar la producció d'un conjunt de plaques fotovoltaïques (DC), per exemple, que generessin una potència de 4,8 KW, que correspondria a un voltatge de 48 V i una intensitat de 100 A ( $P = V \cdot I$ ).



## 4 Parts del monitor d'energia elèctrica

Aquest apartat te com a objectiu explicar detalladament cadascuna de las parts principals del monitor d'energia elèctrica, aquestes contenen imatges que ajuden a entendre millor cada part.

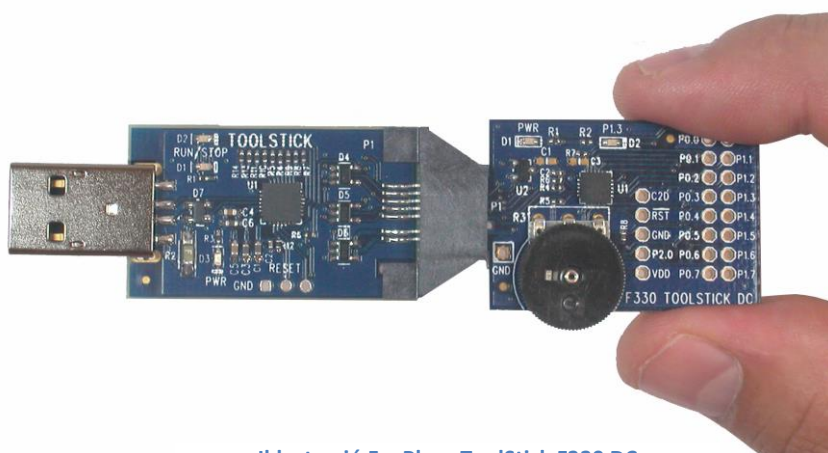
### 4.1 Placa ToolStick F330 DC

Per controlar un procés petit o gran és necessari la utilització d'un microcontrolador, al qual ha d'anar acompanyat d'una sèrie de components per tal de poder funcionar correctament, en aquest cas el microcontrolador escollit és el Intel 8051F330. A l'hora de treballar amb un microcontrolador es poden escollir dos opcions.

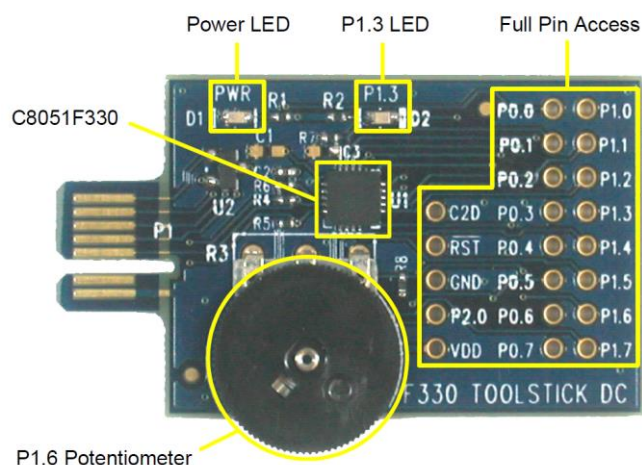
La primera es tracta de treballar amb el xip sol, és a dir, que s'ha de fer un circuit al voltant del microcontrolador perquè funcioni correctament, tenint en compte criteris com: el voltatge d'alimentació, el cristall que s'ha posar perquè treballi a una freqüència determinada o quines s'han d'habilitar/ deshabilitar segons el criteri de funcionament del programador. Si es tria aquesta opció també s'ha de tenir en compte que es necessitarà un placa programadora, que amb l'ajuda d'un software específic ajudarà a gravar els programes en el microcontrolador.

La segona opció es tracta de partir de un circuit ja fet que tingui en compte els criteris abans esmentats, en aquest cas ens referim a utilitzar la placa ToolStick F330 DC. Aquesta placa incorporar un circuit que juntament amb un adaptador USB ajuda a alimentar el microcontrolador fa més accessibles els ports de sortida així com també el reset, la massa del microcontrolador i una tensio de referencia, que es pot utilitzar per a un circuit extern. L'adaptador USB contribueix en l'alimentació del microcontrolador i permet la comunicació entre aquest i un ordinador.

Així doncs per aquest projecte s'escull la segona opció, degut a las facilitats que proporciona ( adaptador USB i circuit per al microcontrolador ) i a més a més en la assignatura Programació de baix nivell: aplicacions industrials dels microcontroladors s'ha après a utilitzar aquesta placa per poder controlar qualsevol procés senzill.

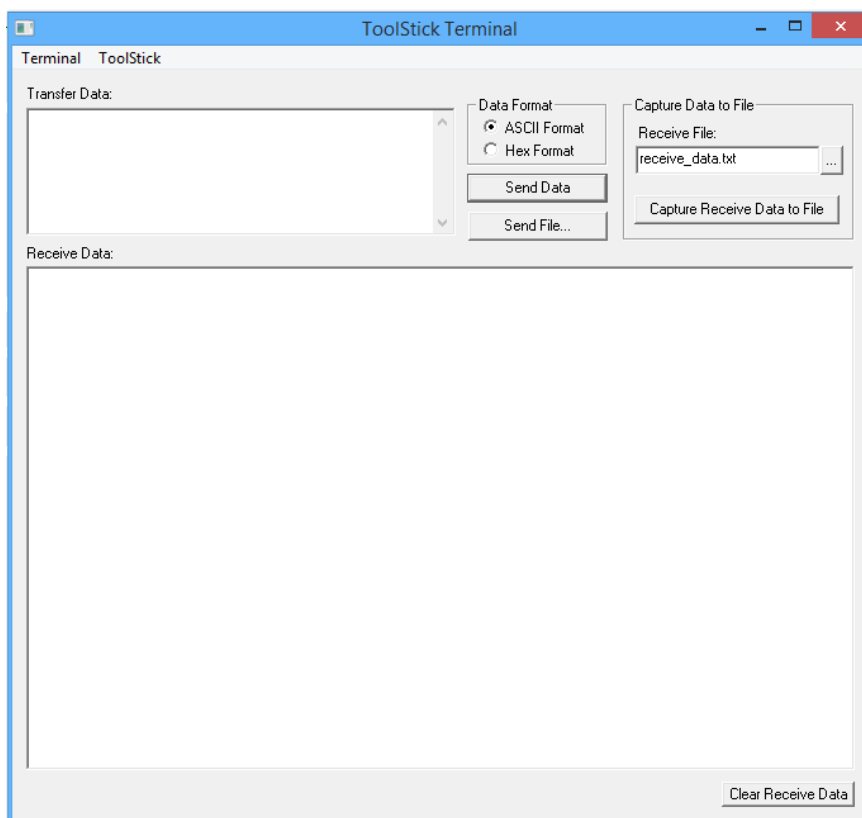


Il·lustració 5 – Placa ToolStick F330 DC



Il·lustració 6 – Circuit on hi incorporat el microcontrolador

El software proporcionat per Silicon Labs permet la fàcil reprogramació del microcontrolador així com un control exhaustiu i en temps real de cada una de les variables que es defineixen el programa. Una opció que proporciona el programa i que serà de molta utilitat en aquest projecte és la ToolStick Terminal. Aquesta opció permet emular un display, finestra anomenada Receive Data, i un teclat, finestra anomenada Transfer Data, amb el que l'usuari pot anar interactuant.



Il·lustració 7 – Eina des de la qual l'usuari rebrà o enviarà informació

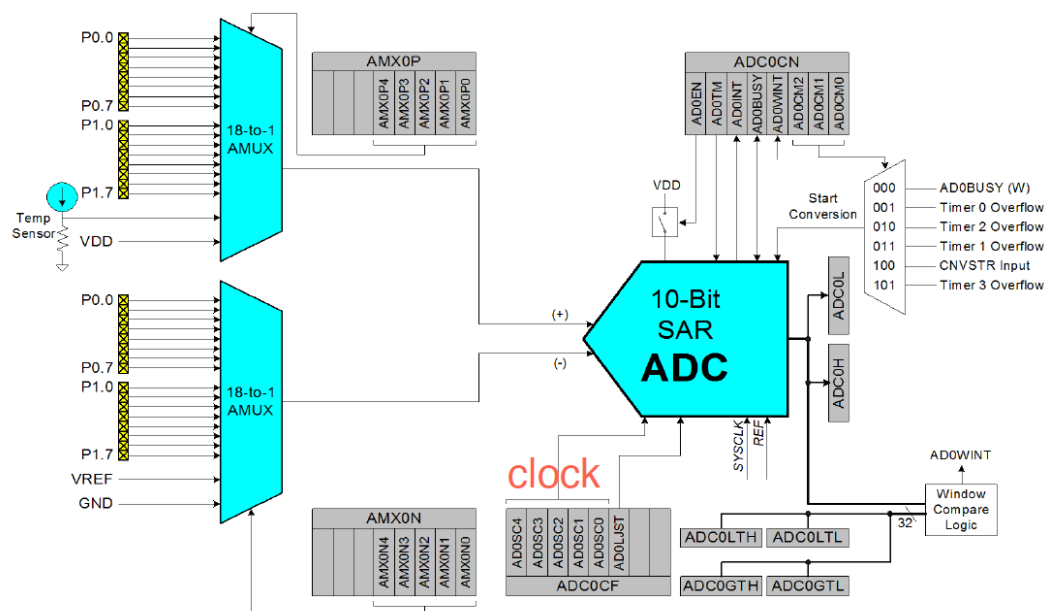
Un aspecte important que s'ha de tenir en compte és el funcionament del convertidor Analògic-Digital, ja que aquest serà l'encarregat d'agafar les mostres dels senyals que es vulgui. El convertidor Analògic-Digital consisteix en dos multiplexors analògics amb 16 entrades totals disponibles per seleccionar les quals poden mostrejar senyals que tinguin una amplitud màxima de 3,4 V aproximadament, amb una velocitat de transferència màxima de 200.000 mostres per segon amb una resolució de 10 bits, és a dir, que és una resolució de:

$$\text{LSB's o nivells} = 2^n \text{ bits} = 2^{10} = 1024 \text{ nivells o LSB's}$$

$$\text{Resolució} = \frac{\text{Voltatge màxim}}{\text{LSB's}} = \frac{3,4}{1024} = 3,30 \text{ mV/LSB}$$

Amb els senyals que es treballi sempre es referenciaran a massa, per tant a la entrada negativa del ADC corresponen al registre AMX0N es posarà massa o GND. Així doncs per l'entrada positiva del ADC, corresponen al registre AMX0P, vindran els senyals que es volem mostrejar. Per engegar i habilitat el ADC s'utilitza el registre ADC0CN, en el que s'especifica amb quin mètode es dona l'ordre per fer l'adquisició d'una mostra d'un senyal, en aquest cas l'ordre s'establirà manualment posant a 1 el registre AD0BUSY.

Quan la conversió s'hagi fet es saltarà a la interrupció del ADC a través del bit AD0INT que es posa a 1 automàticament. Quan s'entra a la interrupció només queda per mirar el resultat de la conversió de la mostra analògica que queda guardat en els registres ADC0H i ADC0L.

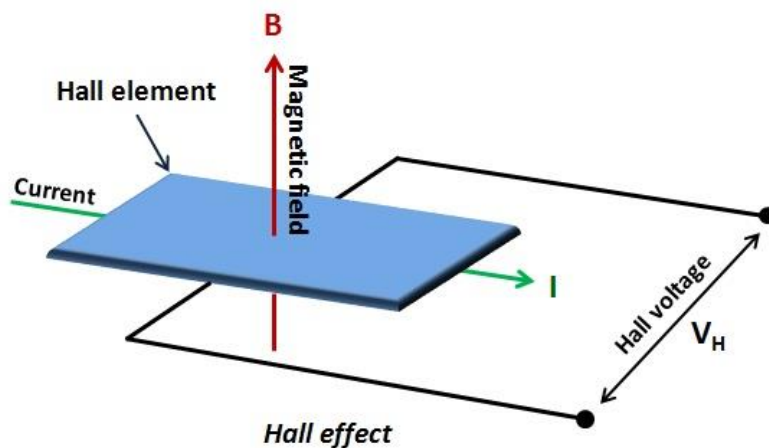


Il·lustració 8 – Esquema funcional del convertidor Analògic-Digital

## 4.2 Sensor de corrent

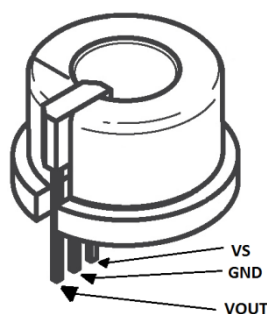
El sensor de corrent que s'utilitza en el prototip és el HONEYWELL CSLT6B100 de llaç obert. El funcionament d'aquests tipus de sensors es basen en l'efecte Hall, que s'explica a continuació.

L'efecte Hall és un fenomen en que quan es fa passar corrent per un conductor dins d'un camp magnètic, aquest exerceix una força transversal sobre les carregues mòbils, que les fan anar cap a un costat del conductor, llavors entre costat i costat del conductor es genera una tensió anomenat tensió de Hall, que es proporcional a la intensitat que circula pel conductor. Com que la tensió de Hall és molt petita, els sensors de Hall el que s'encarreguen es de amplificar aquest voltatge a uns valors manipulables.



Il·lustració 9 – Efecte Hall

La gran avantatge d'aquest tipus de sensors és que no s'ha de tallar el conductor per saber el valor de la intensitat que el travessa. L'únic que s'ha de fer per saber aquest valor és fer passar el conductor pel centre del sensor i conseqüentment dit sensor ens proporciona una valor de tensió que es proporcional a la intensitat que circula pel conductor.



Il·lustració 10 – Sensor de corrent

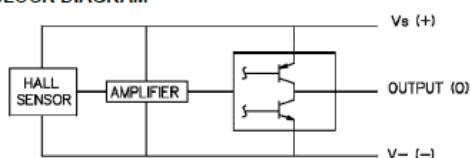
El sensor de corrent té 3 terminals els quals són:

- 1- VS que correspon al valor de tensió amb el que s'ha d'alimentar el sensor.
- 2- GND que correspon a massa.
- 3- VOUT que correspon a la tensió de sortida del sensor, que és proporcional a la intensitat.

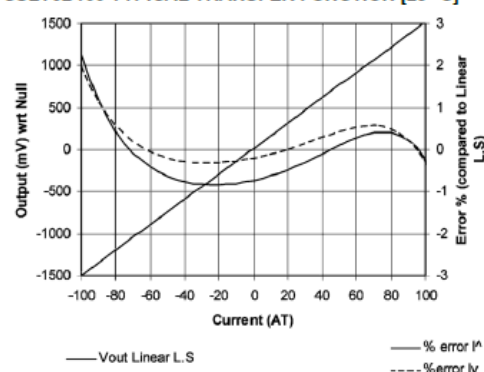
El sensor pot detectar tant corrent continu (des de 0 A fins a 100 A) com corrent altern (des de -100 A fins a 100 A) i té una tensió de sortida de -1,5 V per -100 A i 1,5V per 100 A, depenent de la tensió de referència, és a dir, depèn d'on establim els 0 V. Per saber a on s'han de posar els 0 V de referència, primer s'ha de mirar el voltatge d'alimentació del sensor així com també el corrent d'alimentació i el valor de referència si s'alimenta el sensor a una tensió determinada. Per veure aquests valors s'ha d'observar el datasheet del sensor.

Parameter	Symbol	Min.	Typ.	Max.	Units	Condition
Current range	$I_p$	$\pm 100$	—	—	AT	$< \pm 1.5\%$ error ( $-25\text{ }^{\circ}\text{C}$ to $100\text{ }^{\circ}\text{C}$ [ $-13\text{ }^{\circ}\text{F}$ to $212\text{ }^{\circ}\text{F}$ ])
Supply voltage	$V_s$	4.5	5	10.5	V	—
$V_{out} @ 0\text{ NI}$	$V_o$	2.35	2.5	2.65	V	—
Supply current	$I_s$	—	7	9	mA	no load
Sensitivity	$\Delta V/I$	13.5	16	18.5	mV/AT	$-25\text{ }^{\circ}\text{C}$ to $100\text{ }^{\circ}\text{C}$ [ $-13\text{ }^{\circ}\text{F}$ to $212\text{ }^{\circ}\text{F}$ ]
Hysteresis	—	—	—	0.5	%	$\pm 100\text{ A}$
Temp error - null	$TC_{\Delta V_o/V_o}$	-0.064	—	0.064	%/ $^{\circ}\text{C}$	—
Temp error - gain	$TC_G$	-0.03	—	0.12	%/ $^{\circ}\text{C}$	$-25\text{ }^{\circ}\text{C}$ to $100\text{ }^{\circ}\text{C}$ [ $-13\text{ }^{\circ}\text{F}$ to $212\text{ }^{\circ}\text{F}$ ]
Rise time	$t_r$	—	3	—	$\mu\text{s}$	0 A to 2.0 A

BLOCK DIAGRAM



CSLT6B100 TYPICAL TRANSFER FUNCTION [25 °C]



II·lustració 11 – Datasheet del sensor de corrent

Per fer un bon ús del sensor es miren els valors típics dels principals paràmetres recomanats pel fabricant. Així doncs s'alimentarà el sensor a 5V, que correspondrà a 7 mA de corrent d'alimentació (sense carrega) i a un voltatge de referència de 2,5 V. També s'ha de tenir en compte el gràfic del comportament del sensor, que aquest cas és molt lineal, i l'error en la mesura del sensor, el qual és del 1,5 %.

### 4.3 Comunicació entre el microcontrolador i el PC

En aquest apartat s'explicarà com es comuniquen un microcontrolador amb un PC, tenint en compte els nivells lògics del microcontrolador i del PC, així com també el medi de comunicació, per cable o per aire. Perquè l'explicació quedi més clara es mostraran els diferents esquemes per cada tipus de medi.

#### 4.3.1 Nivells lògics de tensió

L'electrònica digital utilitza sistemes i circuits en què només hi ha dos estats possibles, alt i baix. En els sistemes digitals, les combinacions d'aquests dos estats s'anomenen codis, i s'utilitzen per a representar nombres, símbols, caràcters alfabètics i qualsevol altre tipus d'informació. El sistema de numeració de dos estats es denomina binari i els dos dígitos que s'utilitzen són 1 i 0. Un dígit binari s'anomena bit.

En els circuits digitals, s'utilitzen dos nivells de tensió diferents per representar els dos estats. Un 1 es representa mitjançant un nivell de tensió més elevat, que s'anomena nivell alt, i un 0 es representa mitjançant un nivell més baix de tensió, que s'anomena baix.

Les tensions que s'utilitzen per representar els uns i zeros reben el nom de nivells lògics. L'ideal seria que un nivell de tensió representés el nivell alt i un altre nivell de tensió representés el nivell baix. No obstant això, en un circuit digital pràctic, un nivell alt pot ser qualsevol tensió entre un màxim i un mínim especificats. De la mateixa manera, un nivell baix pot ser qualsevol tensió compresa entre un màxim i un mínim especificats.

Però no tots els sistemes digitals treballen amb els mateixos nivells lògics, per tant s'ha d'observar amb quin tipus de nivell lògic treballa cada sistema, els nivells més comuns són TTL, CMOS, HC per circuits integrats i després hi ha un altre com el RS232 que permet la transferència d'informació bidireccional entre microprocessadors i PC's. En aquest projecte els nivells lògics que es tindran més en compte són el RS232, que és que correspon al port sèrie del PC, i el TTL que és amb el que treballa el microcontrolador Intel 8051F330. A continuació es mostrarà una taula amb els diferents tipus de nivells lògics i les tensions corresponents a cada estat, alt o baix.

Nivells de tensió	TTL	CMOS	HC	RS232
Estat baix (0)	0V a 0.8V	0V a 1.5V	0V a 1V	+3V a +15V
Estat alt (1)	2V a 5V	3.5V a 5V	3.5V a 5V	-3V a -15V

Il·lustració 12 – Taula amb els diferents nivells lògics

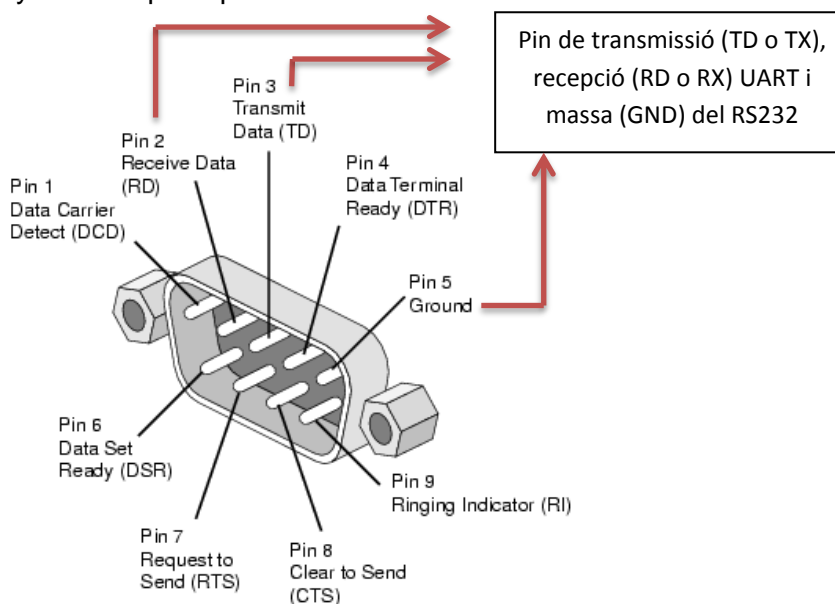
### 4.3.2 Comunicació amb fils

En aquest apartat s'explicarà com s'establirà la comunicació entre el microcontrolador i el PC a través de cablejat. Com s'ha comentat a l'apartat anterior, el microcontrolador i el PC treballen amb diferents tipus de nivells lògics així que es necessitarà com una mena "d'adaptador" que permeti la bona transferència de informació.

Aquest adaptador es tracta d'un circuit integral anomenat MAX232, el qual permet fer una conversió entre els nivells lògics TTL a nivells lògics RS232. Per fer la conversió es necessita que el circuit integrat MAX232 estigui dins un circuit analògic que permeti el cablejat entre el MAX232, el microcontrolador i el PC.

En el circuit s'inclouen cinc condensadors, el valor dels quals determina una velocitat de transferència concreta. Si els condensadors són de 100 nF es pot arribar a una velocitat de transferència de 64 Kilo bits per segon, mentre que si s'utilitzen condensadors de 1µF la velocitat de transferència pot arribar fins 120 Kilo bits per segon.

Per poder fer aquest circuit es necessita que el PC despongui d'un port sèrie així com també d'un allargador DB9 per permetre que es tinguin més accessibles els 9 pins del port sèrie. S'ha de tenir present l'objectiu de cada pin, amb la finalitat de que la transmissió de informació es faci de manera correcta, a continuació es mostrarà la distribució dels pins, senyalant els pins que es faran servir.



Il·lustració 13 – Distribució de pins del port sèrie del PC

Un altre aspecte que s'ha de tenir en compte és la distribució dels diferents ports del microcontrolador amb l'objectiu de poder veure quins ports es reserven per la utilització de la UART o port sèrie. A continuació es mostrarà el diagrama de blocs del Intel 8051F330 per veure la distribució dels seus ports d'entrada/sortida, senyalant els ports i pins que es faran servir.



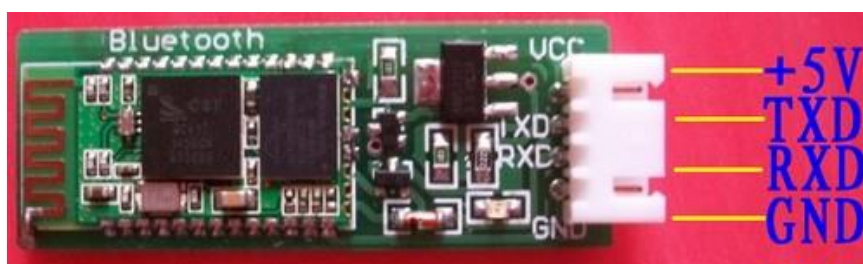




### 4.3.3 Comunicació sense fils

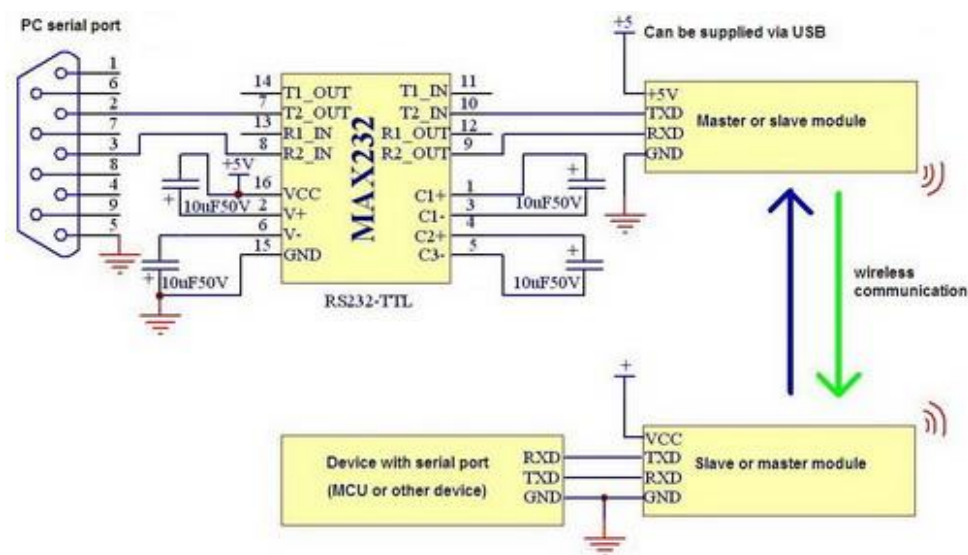
Aquest tipus de comunicació no s'implantarà al prototip d'aquest projecte, però si que s'enunciarà de manera teòrica con s'hauria de fer per implantar-lo. La comunicació sense fils entre el microcontrolador i el PC que es proposa, funciona amb la tecnologia de comunicació anomenada Bluetooth, la qual es basa en la transferència d'informació des de un mòdul transmissor cap a un mòdul receptor per mitja d'ones de radio (radiofreqüència) amb una freqüència fixa de 2,4 GHz.

El nombre de mòduls depèn del PC amb el que pretén comunicar, si aquest disposa de Bluetooth només es necessitarà un mòdul. Si el PC no disposa de Bluetooth, com és el cas, es necessitaran dos mòduls. El mòdul escollit pot funcionar com a receptor (esclau) o com a transmissor (mestre) i amb una velocitat per defecte de 9600 bauds.



## Il·lustració 16 – Mòdul Bluetooth mestre/esclau

La connexió, en aquest cas, seria pràcticament la mateixa que el apartat anterior, la diferència està en que a la sortida del MAX232 aniria el mòdul esclau (receptor) i a la sortida del microcontrolador aniria el mòdul mestre (transmissor). A continuació es mostrarà l'esquema del circuit de transmissió de dades sense fils.



## Il·lustració 17 – Esquema de connexió sense fils entre microcontrolador, PC i MAX232

## 4.4 Circuit analògic

En aquest apartat s'explicarà com s'han dissenyats els diferents subcircuitos que componen el circuit del prototip del monitor d'energia elèctrica d'aquest projecte. Cada part dels subcircuitos s'ha fet amb el software PSIM, i la unió de les diferents parts s'ha realitzat amb el software EAGLE PCB, el qual ajudarà posteriorment a dissenyar la placa de circuit imprès.

### 4.4.1 Disseny del circuit analògic

En aquesta secció s'explicarà el disseny dels diferents subcircuitos dels quals consta el prototip del monitor d'energia elèctrica.

#### 4.4.1.1 Disseny adaptació del senyal de tensió d'entrada

L'escala de la tensió d'entrada d'aquest prototip de monitor d'energia elèctrica treballarà amb una escala principal de 0 fins a 400 volts. Si es treballa amb una senyal de corrent contínua la màxima tensió d'entrada serà de 400 volts, mentre que si es treballa amb una senyal de corrent alternada sinusoidal el valor màxim del senyal d'entrada ha de ser de 400 volts o també seria equivalent dir que el valor eficaç màxim del senyal hauria de ser de:

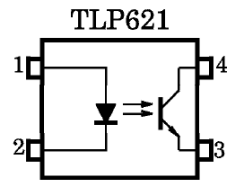
$$\text{Valor eficaç màxim} = \frac{\text{Valor màxim del senyal d'entrada}}{\sqrt{2}} = \frac{400}{\sqrt{2}} = 282,84 \text{ V}$$

Ara s'ha d'observat quin és el rang de tensió d'entrada que accepta la placa ToolStick F330 DC, per posteriorment calcular la reducció de tensió de l'escala de 0 – 400 Volts. Com que es tracta d'una escala de voltatge molt alta, la exactitud es podria veure afectada, per solucionar això es farà una segona escala que anirà de 0 fins a 60 volts.

La placa accepta una un rang de tensió d'entrada comprès entre 0 i 3,38 Volts aproximadament. Per no treballar amb un valor màxim de voltatge amb decimals s'ha determinat que el valor màxim que entrarà a la placa sigui de 3 volts, és a dir, que l'escala de treball del prototip serà de 0 fins a 3 volts.

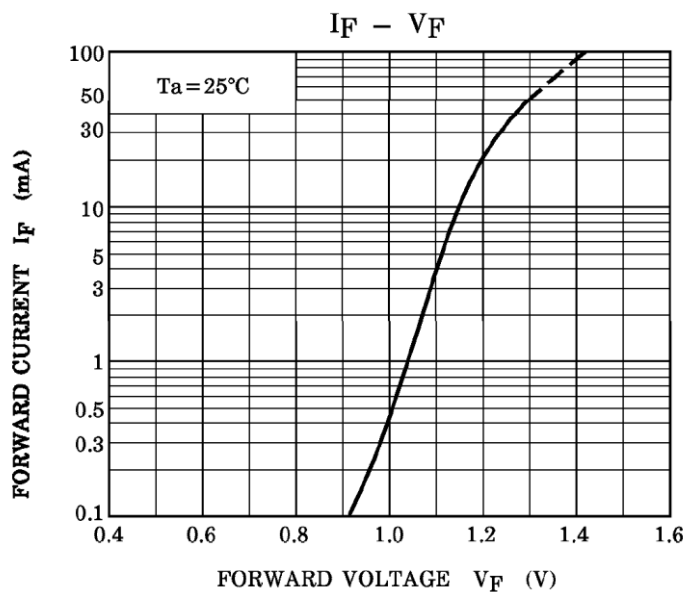
Una vegada arribats a aquest punt s'ha de pensar com es pot protegir elèctricament la placa ToolStick F330 DC, ja que les tensions d'entrada són molt altes i es podria donar el cas que hi hagués un curt circuit, fet que espatllaria la placa. Per fer aquesta tasca s'utilitzarà un optoacoblador.

Un optoacoblador és un component electrònic que s'utilitza com a transmissor i receptor òptic (de llum), és a dir poden transmetre d'un punt a un altre un senyal elèctric sense necessitat de connexió física ni cables (per l'aire), mitjançant un senyal lluminós. Com a transmissor òptic s'utilitza un díode LED i con a receptor òptic s'utilitzen fototransistors o triacs. La tensió de sortida de l'optoacoblador és proporció al a la tensió que se li aplica al díode LED, cal comentar que aquest comportament lineal només es dona en un petit rang de tensió. L'optoacoblador que s'utilitzarà en el prototip és un Toshiba TLP621, que té com a receptor òptic un fototransistor. Per saber aquest rang de tensió al qual treballa de forma lineal, s'ha d'anar a consultar el datasheet del component.



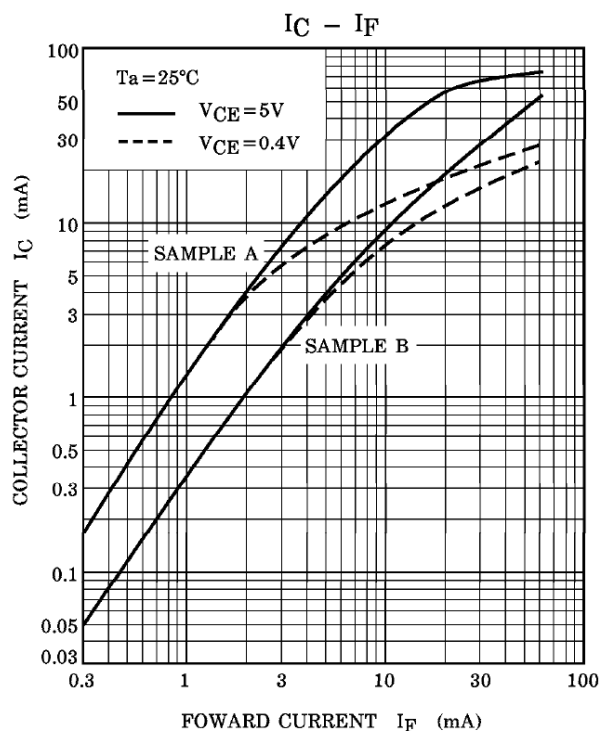
- 1 : ANODE
- 2 : CATHODE
- 3 : EMITTER
- 4 : COLLECTOR

II·lustració 18 – Distribució de potes de l'optoacoblador TLP621



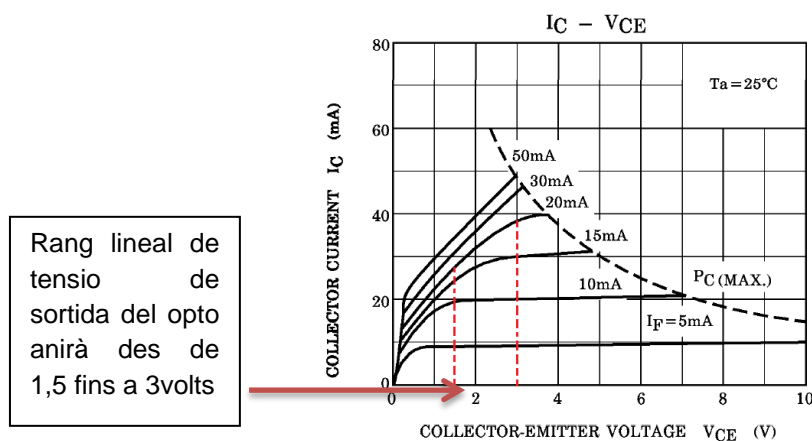
El rang de tensió lineal aplicada al LED va des de 0.8 fins a 1.2 V. I el rang de corrent lineal aplicat al LED va des de 0.1 fins a 20 mA.

II·lustració 19 – Relació entre voltatge ( $V_F$ ) i corrent ( $I_F$ ) que se li pot aplicar al LED



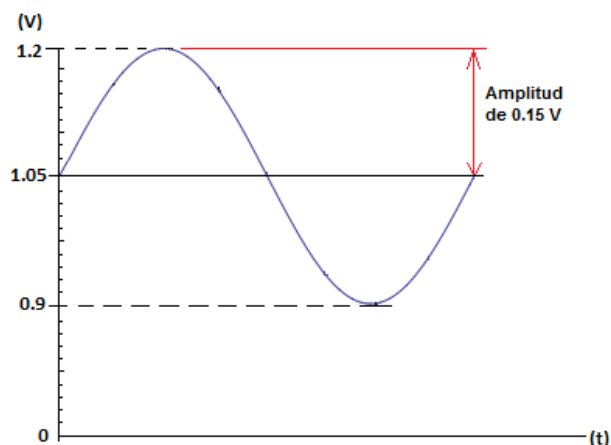
El rang de corrent lineal aplicat al LED va des de 0.1 fins a 20 mA. I el rang lineal de corrent al col·lector va des de 0.1 fins a 30 mA

II·lustració 20 – Relació entre corrent ( $I_F$ ) que se li pot aplicar al LED i corrent al col·lector ( $I_C$ )



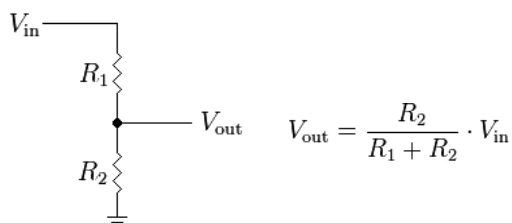
Il·lustració 21 – Relació entre corrent al col·lector ( $I_C$ ) i tensió a la sortida del optoacoblador ( $V_{CE}$ )

Con que el rang de treball lineal de l'optoacoblador va des de 0.9 fins a 1.2 volts, la senyal d'entrada de 400 volts s'ha de reduir fins a 0.15 volts i a més se li ha d'afegir un nivell de continua de 1.05 V, així la forma d'ona que s'ha obtenir, si tenim com a entrada una senyal sinusoïdal de 400 volts d'amplitud, és la següent:



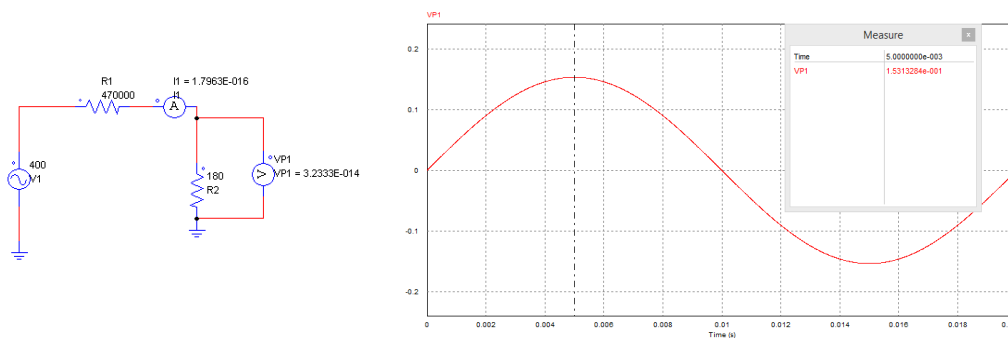
Il·lustració 22 – Forma del senyal que s'ha d'introduir al opto, quan es treballi amb un senyal d'entrada de 400 volts d'amplitud

A continuació es farà el divisor de tensió a on la tensió d'entrada ( $V_{in}$ ) seran 400 volts i la tensió reduïda ( $V_{out}$ ) seran els 0,15 volts.



Il·lustració 23 – Esquema i formula d'un divisor de tensió

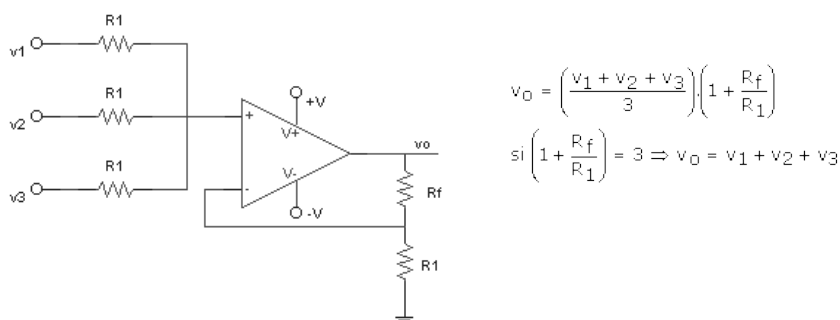
Es fixa la  $R_1$  de 470 K $\Omega$  i llavors es troba la  $R_2$  que en aquest cas dona 180  $\Omega$ .



II-lustració 24 – Esquema i senyal del sortida del divisor de tensio fet amb el programa PSIM

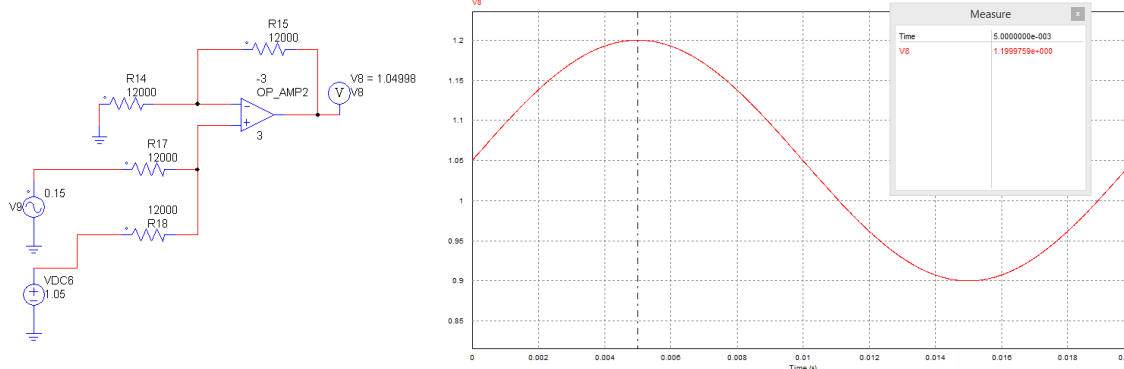
La senyal que s'ha d'introduir a l'optoacoblador (il·lustració 22) ,està composta per una senyal sinusoidal i un senyal continua, la suma d'aquestes dos s'ha de fer mitjançant un amplificador operacional. D'aquest se'n utilitzaran 3, l'amplificador operacional 1 que serveix per sumar el senyal sinusoidal (0,15 V d'amplitud) i el senyal continu (1,05 V), l'amplificador operacional 2 que serveix per transformar els valors inferiors o iguals a 60 V, que correspon a valors per sota o iguals de 0,0225V (escala 0-400 v),en valors corresponents a l'escala de 0-60V, per exemple, transforma els 0,0225 V(que serien 60 V en l'escala de 400 V) en 1,5 V (que serien els 60 V corresponents a l'escala de 60 V). Per últim l'amplificador operacional 3 que serveix per sumar el senyal generat a l'amplificador operacional 2 i un senyal continu de 1,5 V (se sumen aquest 2 senyals perquè posteriorment el microcontrolador pugui fer el càlcul de l'escala de 0 a 60 volts correctament ).

En l'amplificador operacional 1 el muntatge que es farà servir és un sumador no inversor, el qual es mostra a continuació.



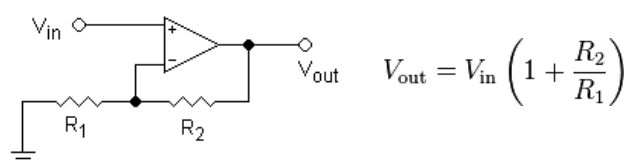
II-lustració 25 – Esquema i formula d'un amplificador operacional sumador no inversor

En aquest cas el que es farà serà posa  $R1=Rf$ , ja que només s'estan sumant 2 senyals.



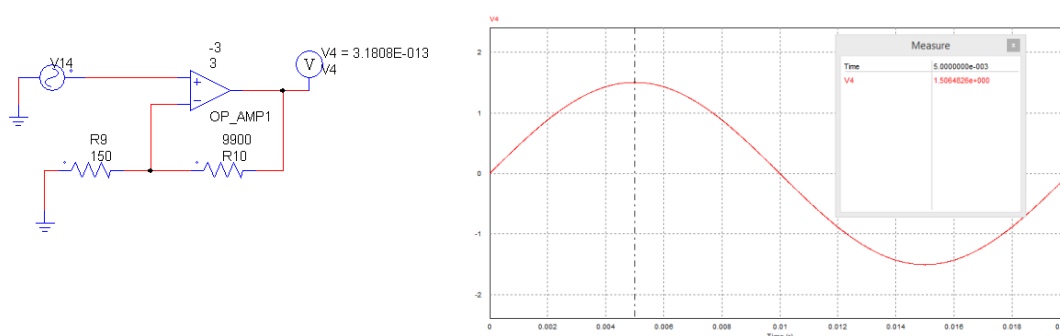
II-lustració 26 – Esquema i senyal del sortida del amplificador sumador no inversor fet amb PSIM

En l'amplificador operacional 2 el muntatge que es farà servir és un no inversor, el qual es mostra a continuació.



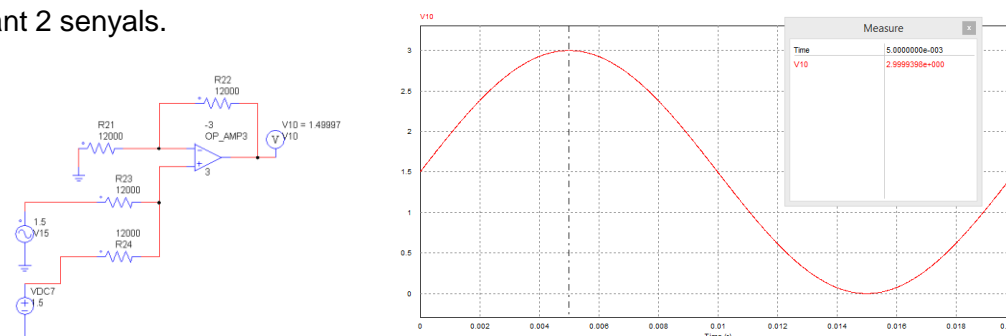
Il·lustració 27 – Esquema i formula d'un amplificador operacional no inversor

En aquest cas la tensió d'entrada màxima és 0,0225 V (Vin) i la tensió de sortida serà 1,5V (Vout), es fixa la R1 a 150 Ω i llavors es troba R2, que en aquest cas dona 9,9 KΩ (s'utilitzarà un potenciòmetre de 10 KΩ)



Il·lustració 28 – Esquema i senyal del sortida del amplificador no inversor fet amb PSIM

En l'amplificador operacional 3 el muntatge que es farà servir és un sumador no inversor. L'esquema i la seva formula s'ha mostrat anteriorment, pel que fa al valor de RF es posarà igual R1, igual que l'amplificador operacional 1, aquí només s'estan sumant 2 senyals.



Il·lustració 29 – Esquema i senyal del sortida del amplificador sumador no inversor fet amb PSIM

Tornant a l'optoacoblador, si es treballés amb un senyal d'entrada sinusoidal de 400 volts d'amplitud, a la sortida del optoacoblador s'hauria de tenir un senyal de 1.5 volts d'amplitud i un nivell de continua de 1,5 volts.

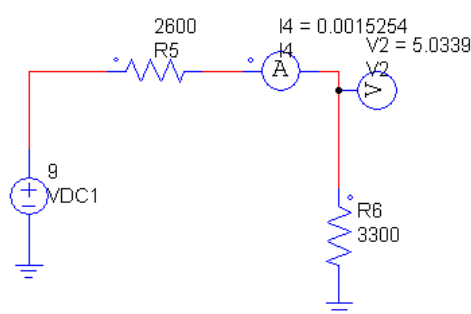
Per obtenir això el que es farà serà alimentar el optoacoblador a 3 volts i amb un potenciòmetre entre emissor (del fototransistor) i massa s'ajustarà per obtenir els 1,5 volts de continua. Per trobar el valor que ha de tenir el potenciòmetre s'anirà a consultar el rang de corrent al col·lector:



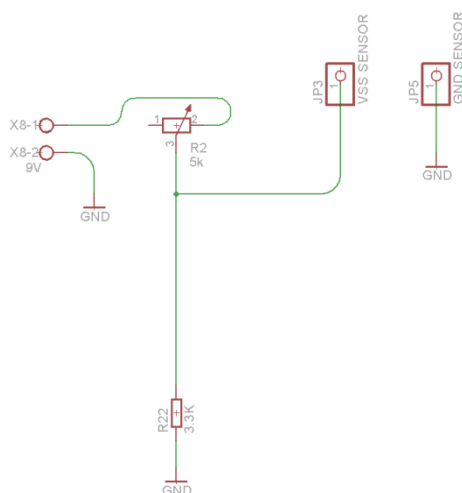
#### 4.4.1.2 Disseny del circuit per alimentar el sensor de corrent

A l'apartat 4.2 es va parlar del sensor de corrent que s'utilitzaria en el prototip del monitor d'energia d'aquest projecte, el HONEYWELL CSLT6B100. A la il·lustració 11 (pàgina 16) es mostrava una part del datasheet del sensor a on es veia reflectit la tensió d'alimentació que necessitava el sensor per funcionar correctament, aquesta tensió és de 5 volts. Per obtenir aquest 5 volts es farà servir un divisor de tensió, del qual s'ha mostrat el seu esquema i formula (il·lustració 23, pàgina 23). Per poder fer-lo s'utilitzarà una pila de 9 volts, que juntament amb un potenciòmetre i una resistència es podrà obtenir el 5 volts amb els que s'alimentarà el sensor de corrent.

En aquest cas la R2 es fixarà a 3,3 K $\Omega$  i llavors es troba la R1 que en aquest cas donarà 2,6 K $\Omega$ , com que aquest valor no està normalitzat es farà servir un potenciòmetre de 5 K $\Omega$ .



Il·lustració 32 – Esquema del circuit per alimentar el sensor de corrent fet amb PSIM



Il·lustració 33 – Esquema del circuit per alimentar el sensor de corrent fet amb EAGLE PCB

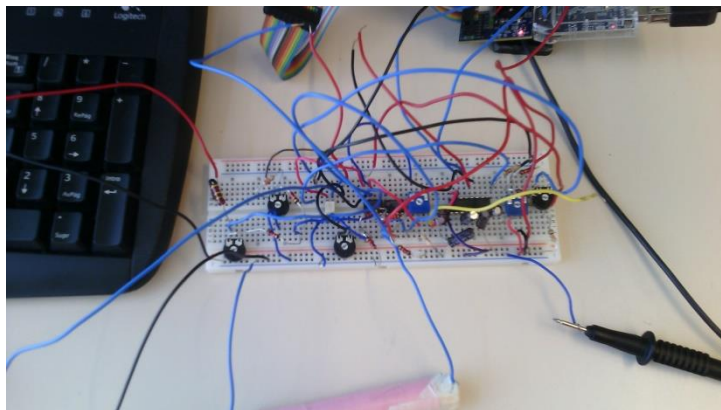
Al esquema fet amb EAGLE s'inclouen els pins o pinheads, que tenen per funció:

- Fer arribar l'alimentació al sensor de corrent (pin JP3, VSS SENSOR).
- Agafar la massa del sensor de corrent (pin JP5, GND).



#### 4.4.2 Disseny de la placa de circuit imprès (PCB)




Els subcircuitos analògics anteriorment enunciats primerament es van muntar sobre una placa protoboard, realitzant les connexions a traves de cables. Es van fer proves sobre aquesta placa, però hi havia vegades en que les proves no funcionaven correctament degut a que els cables es podien moure fàcilment, fet que provocava que no hi hagués contacte elèctric. A part d'això el fet de muntar el circuit a la protoboard donava un aspecte desordenat i si cap persona volia veure com anaven les connexions se li feia difícil entendre-les.

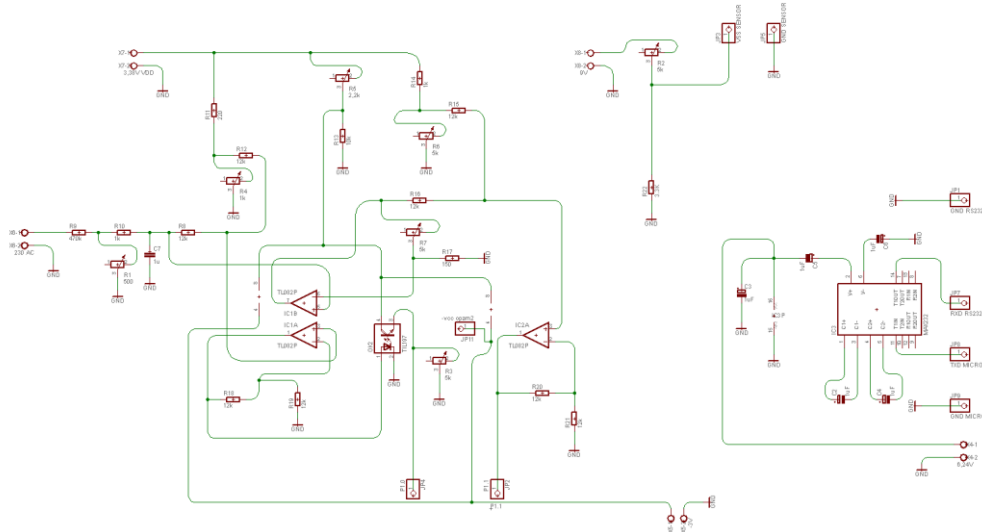


Il·lustració 34 – Circuit analògic sencer muntat sobre la placa protoboard


A partir d'aquí es va decidir tornar muntar el circuit però sobre una placa de circuit imprès PCB.

Per fer el disseny de la PCB s'ha utilitzat el software de lliure llicència, EAGLE PCB. A continuació es defineixen els passos que s'han seguit per poder fer el disseny de la PCB.

- 1- Crear un projecte per guarda tots els arxius, File → New → Ubicació → Save.
- 2- Clicar sobre la icona Add  i buscar tots els components que es faran servir.
- 3- Una vegada es tenen tots el components, es començarà a realitzar l'esquema elèctric del circuit. Per fer les connexions entre els components es clica la icona Wire , per fer 2 o més connexions en un mateix punt es clica la icona Junction , si es vol puntejar cables es poden passar un sobre altres.
- 4- Quan es fan totes les connexions s'ha d'assegurar que els components que ho requereixin tinguin valor i nom.





II-lustració 35 – Esquema elèctric complet del circuit analògic


- 5- Es clica la icona Generate to board  per començar a distribuir els components (que mantenen les connexions entre ells) sobre la PCB. Con que es traca d'una versió gratuïta, les dimensions de la PCB màximes poden ser de 100x80 mm. El programa té una opció que organitza automàticament els components, però aquest opció no és molt recomanable, ja que no aprofita ve l'espai i a vegades et fa fer plaques de dobles cara per circuits senzills i que no tenen molta densitat de components.



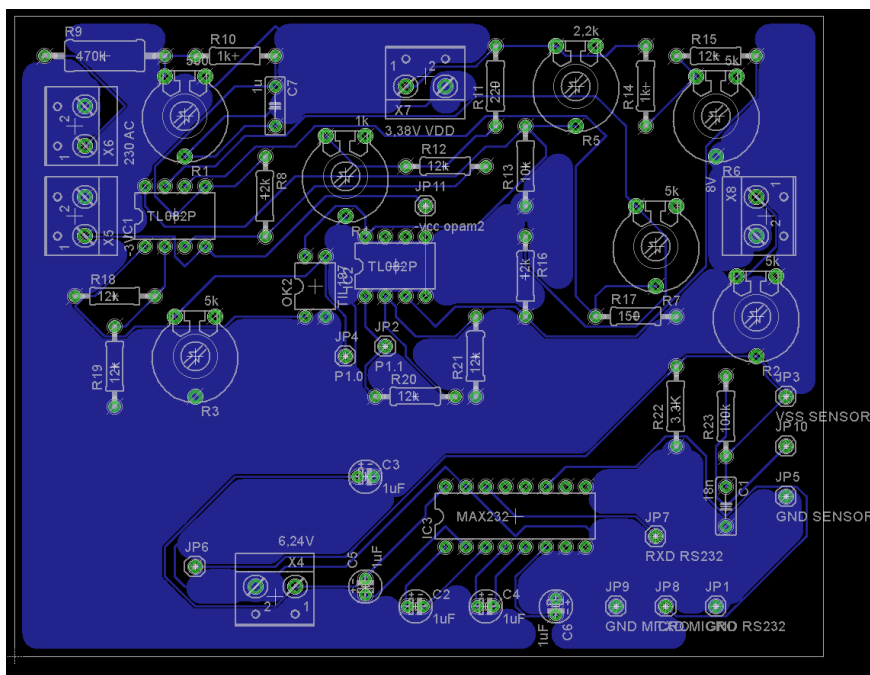
II-lustració 36 – Realització del disseny de la PCB amb el programa EAGLE

6- Es clica la icona Route  i es van fent cada via manualment.

7- Ara es crearà una capa de massa, clicant la icona Polygon  i marcant el marge de la PCB, després es clica botó dret sobre el contorn que s'ha creat i es clica la opció Name i es canvia per GND.

8- Per veure com a quedat la capa de massa es clica la icona Ratsnest  .

A continuació es mostra el resultat del disseny de la PCB del prototip del monitor d'energia d'aquest projecte.



Il·lustració 37 – Resultat del disseny de la PCB del monitor d'energia elèctrica

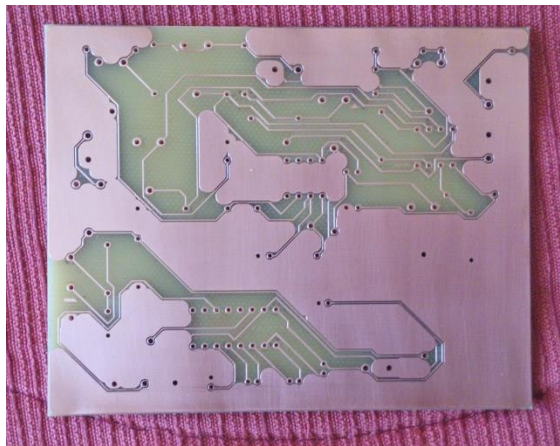
Ara es comentarà amb quina eina es realitza el disseny de la PCB a la realitat. La fabricació de PCB's es fa mitjançant una màquina anomenada Ploter fresador, que realitza tant les pistes com els forats de la PCB.



Il·lustració 38 – Ploter fresador que serveix per la fabricació de PCB's

Perquè la màquina entengui la distribució de pistes i forats, se li envien uns arxius anomenats Gerber que es generen a partir del projecte creat amb el programa EAGLE.

Per últim es mostrarà el resultat de la PCB física sense components i posteriorment amb tots el components.



Il·lustració 39 – PCB sense components



Il·lustració 40 – PCB amb tots els components

## 5 Programació

L'objectiu d'aquest apartat és explicar tots els codis de programació que s'han utilitzat en aquest projecte. Cal mencionar que el lector ha de tenir uns coneixements de programació bàsics, per tal d'entendre l'explicació correctament. El microcontrolador es programarà amb llenguatge C i l'adquisició de dades es programarà amb un llenguatge visual gràfic.

### 5.1 Programació del microcontrolador

#### 5.1.1 Explicació del codi

En aquest apartat s'explica detalladament quina és la funció de cada part del codi, per poder entendre com es realitzen els càlculs i com es genera la informació, que posteriorment s'envia cap al PC. Per entendre millor el funcionament del codi es separa el codi a la part esquerra i a la dreta s'ubica l'explicació de cada tros de codi que queden limitats per claudàtors.

```
#include <C8051F330.h>
#include <stdlib.h>
#include <stdio.h>
#include <math.h>

#define PI (3.14159)
#define SYSCLK 24500000
#define BAUDRATE 9600
#define OSC_PER_INST (12)

#define INTERRUPT_Timer_2_Overflow 5
#define INTERRUPT_ADC 10

typedef unsigned char tByte;
typedef unsigned int tWord;
typedef unsigned long tLong;

tWord resultatlsb,Inc,numinterrupts,ko;
tByte contador=0,index=0,mosampli=0,rep=0,
      reppotenci=0, numsenyal=0,vegades8=0;

float periode,frecuencia,tensio,potencia,
      potenciaKWH, periodemostres,vat,acvat,
      de400v,vef400v,amplitu400v,de60v,vef60v,
      amplitu60v,deintensi,vefintensi,amplituintensi,
      t,volts,acumulapotencia,tarifa,potencimax,preu;

bit confir=0,bloqueigtarif,bloqueireresult,
    bloqueigconver, bloqueigpotencia,
    alarmapotenmax,rebrepotencia,rebretarif;
```

Primer s'han de posar les llibreries que faran falta en el projecte com la `stdio.h` o la `stdio.h` on hi ha funcions que ens ajudem a fer la conversió de string a float i viceversa, o funcions que ens ajuden a realitzar càlculs matemàtics com la `math`. Després es posa les constants que faran falta com la freqüència del sistema, el baudrate o les oscil·lacions per instrucció. Seguidament es defineixen les interrupcions dels perifèrics que s'utilitzarà, en aquest cas s'utilitzen el timer 2 i el convertidor Analògic-Digital. Després es defineixen els typedefs per comoditat a l'hora de crear variables. Després es defineixen totes les variables que s'utilitzarà en el programa. En aquest cas es defineixen unes variables de tipus int i char per emmagatzemar comptatges, float per emmagatzemar càlculs, bit que serveixen com a controls de seqüència, per exemple per saber si s'han d'introduir dades o si s'han de mostrar resultats per la UART.

```

char vectarif [4];
char vecmaxpotencia [5];
char vecpotencia [6];
char vecenergia[9];
char vecpreu[9];

void PCA_Init()
{
    PCA0MD  &= ~0x40;
    PCA0MD  = 0x00;
}

void ADC_Init()
{
    switch (numsenyal)
    {
        case 0:
            AMX0P  = 0x08;
            break;
        case 1:
            AMX0P  = 0x09;
            break;
        case 2:
            AMX0P  = 0x0A;
            break;
    }
    AMX0N  = 0x11;
    ADC0CN = 0xC0;
}

```

Després es defineixen uns vectors de tipus char, que serviran per manipular les dades que introdueix l'usuari i el resultats dels càlculs. Seguidament es comencen a definir totes les funcions del programa. Primer es comença amb la funció **void PCA\_Init** que desactiva el watchdog timer, perquè no es provoqui un reset al programa. Després amb la funció **void ADC\_Init** s'estableixen els paràmetres per configurar el convertidor Analògic-Digital. Aquí se li diu al micro que quan la variable numsenyal canviï de valor, que vagi canviant l'entrada positiva del ADC (Analogic Digital Converter) a través del AMX0P (0x08 és el port P1.0, 0x09 és el port P1.1 i 0x0A és el port P1.2). AMX0N serveix per establir l'entrada negativa, que en aquest cas és GND. El ADC0CN serveix per activar el ADC.

```

void Port_IO_Init()
{
    // P1.0 - Skipped,   Open-Drain, Analog
    // P1.1 - Skipped,   Open-Drain, Analog
    // P1.2 - Skipped,   Open-Drain, Analog
    XBR0  = 0x01; // Enable UART0
    XBR1  = 0x40;
    P1MDIN = 0xF8;
    P1SKIP = 0x07;
}

void Oscillator_Init()
{
    OSCICN = 0x83;
}

```

En la funció **void Port\_IO\_Init** s'habiliten els ports que s'utilitzaran, a través de P1MDIN i P1SKIP. Així com també s'habilita la UART0 (Port sèrie) a través de XBR0 i XBR1.

En la funció **void Oscillator\_Init** s'ajusta el microcontrolador perquè vagi a màxima velocitat (24,5 Mhz) a través de OSCICN.



```

void Interrupts_Init()
{
    IE      = 0x80;
    EIE1    = 0x08;
}

void sEOS_Init_Timer2( const float freqmostreig)
{
    tWord Reload_16;
    tByte Reload_08H, Reload_08L;
    Inc=0;
    TMR2CN=0x04;
    Inc=
    freqmostreig/(1/(float)(SYSCLK/OSC_PER_INST));
    Reload_16=(tWord)(65536-Inc);
    Reload_08H=(tByte)(Reload_16 / 256);
    Reload_08L=(tByte)(Reload_16 % 256);
    TMR2H      = Reload_08H;
    TMR2RLH    = Reload_08H;
    TMR2L      = Reload_08L;
    TMR2RLL    = Reload_08L;

    ET2=1;
    TR2=1;
    EA=1;
}

```

La funció **void Interrupts\_Init** serveix per habilitar totes les interrupcions, amb IE, i per activar les interrupcions del ADC a través de EIE1.

En la funció **void sEOS\_Init\_Timer2** serveix per configurar el timer 2, perquè generi interrupcions cada x temps que es vulgui. La avantatge d'aquest timer és que és de carrega automàtica, és a dir, que quan acabi el comptatge, ell mateix sap a partir de quí punt a de tornar a començar a comptar. El que es fa aquí és calcular quina constant de temps (Inc) s'han de posar als registres del timer per aconseguir una interrupció cada x temps que es vulgui (freqmostreig). Com que el timer es de 16 bits i els registres del timer són de 8 bits (TMR2H i TMR2L) el que es fa és treballar amb una variable de 16 bits (Reload\_16) i llavors es parteix en dos (Reload\_08H i Reload\_08L) i així poder posar els valors als registres del timer.

```

void UART0_Init (void)
{
    SCON0 = 0x10;
    if (SYSCLK/BAUDRATE/2/256 < 1)
    {
        TH1 = -(SYSCLK/BAUDRATE/2);
        CKCON &= ~0x0B;
        CKCON |= 0x08;
    }
    else if (SYSCLK/BAUDRATE/2/256 < 4)
    {
        TH1 = -(SYSCLK/BAUDRATE/2/4);
        CKCON &= ~0x0B;
        CKCON |= 0x01;
    }
    else if (SYSCLK/BAUDRATE/2/256 < 12)
    {
        TH1 = -(SYSCLK/BAUDRATE/2/12);
        CKCON &= ~0x0B;
    }
    else

```

La funció **void UART0\_Init** serveix per configurar la UART o port sèrie. En aquest cas el que es fa és utilitzar el timer 1 en mode UART, a través del registre TMOD i se li carrega als registres d'aquest un valor que es calcula i que va en funció de la velocitat de rellotge del microcontrolador (SYSCLK) i de la velocitat de transmissió de bits (BAUDRATE). Per últim es posa en marxa el timer a través del registre TR1 i es posa a 1 el registre TI0 per indicar que es pot començar a transmetre caràcters pel port sèrie.

```

{
    TH1 = -(SYSCLK/BAUDRATE/2/48);
    CKCON &= ~0x0B;
    CKCON |= 0x02;
}
TL1 = TH1;
TMOD &= ~0xf0;
TMOD |= 0x20;
TR1 = 1;
TI0 = 1;
}

```

**void sEOS\_ISR(void) interrupt INTERRUPT\_Timer\_2\_Overflow**

```

{
    TF2H=0;
    numinterrups++;
    if (numinterrups==10000)
    {
        numinterrups=0;
        acumulapotencia=acumulapotencia+(potencia/3600);
        if (acumulapotencia<0)
        {
            acumulapotencia=0;
        }
        sprintf(vecenergia,"%f",acumulapotencia);
        preu=acumulapotencia*tarifa;
        if (preu<0)
        {
            preu=0;
        }
        sprintf(vecpreu,"%f",preu);
        potencia=potencia*1000;
        ko=potencia;
        potencia=ko/1000.0;
        if (potencia<0)
        {
            potencia=0;
        }
        sprintf(vecpotencia,"%f",potencia);
        bloqueireult=0;
    }
    t=t+periodemostres;
    if (t==periode)
    {
        t=0;
    }
    AD0BUSY=1;
}

```

Per calcular la energia, la potència que es calcula ve donada en KW i per passar-la a KWh es divideix per 3600 segons que és una hora.

sprintf és una funció que transforma un nombre amb decimals a un vector. Dita funció es troba en la llibreria stdio.h.

La funció **void sEOS\_ISR(void) interrupt**

**INTERRUPT\_Timer\_2\_Overflow**

s'executa cada x temps que es provoca una interrupció. TF2H es posa a zero perquè torni a entrar a la interrupció quan sigui el moment, és a dir, 0,1 ms. Com es va dir al principi es treballarà amb senyals alternes de 50 Hz i es vol dividir el període del senyal en 200 rectangles, el que comportava que cada vegada s'hagués d'agafar 1 mostra cada 0,1 ms (base dels rectangles, càlcul fet a la pàgina 8), és a dir, que les interrupcions es provocaran cada 0,1 ms. Com que es vol calcular i actualitzar els valors de energia i preu cada segon, el que es fa és anar incrementant la variable numinterrups fins que arribi a 10000, és a dir, que es tindrà un segon cada 10000 interrupcions. Així mateix quan arribi a aquest valor voldrà dir que ja tenim els 200 rectangles i per tant ja es té un període i podem posar a zero la variable t per tornar a compta un altre període. Per últim posem AD0BUSY a 1 per indicar que ja han passat 0,1ms i que el ADC ja pot anar a agafar una mostra del senyal.



```

void RSI_ADC(void) interrupt INTERRUPT_ADC
{
    AD0INT =0;
    if(numsenyal==0)
    {
        vegades8++;
        resultatlsb= ((ADC0H <<8)+ADC0L);
        volts=(resultatlsb*0.00330078125)-1.5;
        if (volts<0)
        {
            volts=0;
        }
        if(vegades8<=200)
        {
            vat=volts*volts*periodemostres;
            acvat=acvat+vat;
            if (vegades8==200)
            {
                vegades8=0;
                de400v=acvat/periode;
                vef400v=sqrt(de400v)*266.67;
                amplitu400v=1.414213562*vef400v;
                if (amplitu400v<60)
                {
                    numsenyal++;
                    acvat=0;
                    ADC_Init();
                }
                if (amplitu400v>60)
                {
                    numsenyal=2;
                    acvat=0;
                    tensio=vef400v;
                    ADC_Init();
                }
            }
        }
    }
    if(numsenyal==1)
    {
        vegades8++;
        resultatlsb= (ADC0H <<8)+ADC0L;
        volts=(resultatlsb*0.00330078125)-1.5;
        if (volts<0)
        {
            volts=0;
        }
    }
}

```

La funció **void RSI\_ADC(void) interrupt INTERRUPT\_ADC** serveix per fer els càlculs de tensió, intensitat i potència, a partir de l'adquisició de mostres del senyal de tensió d'entrada i del senyal del sensor de corrent. AD0INT es posa a zero perquè torni a entrar a la interrupció quan sigui el moment, és a dir, quan a la interrupció del timer 2 s'executi la instrucció AD0BUSY=1 que indica que s'ha d'agafar una mostra del senyal. La variable numsenyal serveix per identificar quin senyal s'està mostrejant. numsenyal = 0 (P1.0), indica que s'està mostrejant el senyal de tensió d'entrada que correspon a l'escala de 0 fins a 400 V. numsenyal = 1 (P1.1), indica que s'està mostrejant el senyal de tensió d'entrada que correspon a l'escala de 0 fins a 60 V. numsenyal = 2 (P1.2), indica que s'està mostrejant el senyal del sensor de corrent que correspon a l'escala de -100 fins a 100 A. La variable vegades8 serveix per anar comptant les adquisicions. En els registres ADC0H i ADC0L es guarda el resultat de la conversió en LSB's. En la variable volts es guarda el resultat de la conversió en volts. Mentre vegades8 sigui igual o menor a 200, en la variable vat es fa part del càlcul de fórmula del valor eficaç per cada adquisició després aquest càlcul es van acumulant en la variable acvat i quan vegades8 sigui igual a 200, es reinicialitza vegades8, i s'acaba de fer l'altre part del càlcul del valor eficaç. En amplitu400v es guarda el valor màxim de la senyal d'entrada. Posteriorment es mira el valor d'amplitud màxima i si aquest valor és inferior a 60 V es torna a fer un altre adquisició perquè el valor de tensió doni més exacte ja que passaria d'una escala de 0 – 400 V a una de 0 – 60 V. Aquest canvi d'escala es fa gracies a la funció **ADC\_Init()** que canvia l'entrada positiva del ADC de P1.0 a P1.1. Si el valor és superior a 60 V es deixa com a bo el valor de tensió calculat i saltem a tractar la intensitat (numsenyal=2).

```

if(vegades8<=200)
{
    vat=volts*volts*periodemostres;
    acvat=acvat+vat;
    if (vegades8==200)
    {
        vegades8=0;
        de60v=acvat/periode;
        vef60v=sqrt(de60v)*40;
        amplitu60v=1.414213562*vef60v;
        numsenyal++;
        acvat=0;
        tensio=vef60v;
        ADC_Init();
    }
}
}

```

Quan numsenyal =1 (P1.1, escala 0 – 60 V), el procés és el mateix, s'agafa la conversió en LSB'S i es guarda en resultatlsb i després es passa a volts (els 0.00330078125 són volts/LSB) i es guarda en la variable volts (a la placa li arriba un senyal sinus de 0 fins a 3 V i es vol treballar amb un senyal de -1.5 fins a 1.5 V així que cada mostra se li resta 1.5 V), després es calcula vat i s'acumula el resultat a acvat, quan s'arriba a vegades8=200 es calcula el valor eficaç i màxim del senyal. Una vegada es fa això s'actualitza el valor de tensió i s'augmenta el valor de numsenyal en 1 per obtenir numsenyal==2 i passar a tractar el senyal del sensor de corrent i consegüentment calcula la intensitat .

```

if(numsenyal==2)
{
    vegades8++;
    resultatlsb= (ADC0H <<8)+ADC0L;
    volts=(resultatlsb*0.00330078125)-2.535;
    if (volts<0)
    {
        volts=0;
    }
    if(vegades8<=200)
    {
        vat=volts*volts*periodemostres;
        acvat=acvat+vat;
        if (vegades8==200)
        {
            vegades8=0;
            deintensio=acvat/periode;
            if (deintensio<0.000380)
            {
                deintensio=0;
            }
            vefintensio=sqrt(deintensio)*117;
            amplituintensio=1.414213562*vefintensio;

```

La constant es troba a partir del valor de l'arrel quadrada de la variable deintensio i del valor vefintensio que hauria de donar per una determinada potència.

Quan es tracta el senyal del sensor de corrent la conversió en LSB's és la mateixa, però per fer la conversió en volts s'ha de tenir en compte que la tensió de referència 0 V és de 2,50 V, els decimals de més són perquè a la practica no es pot ajustar exactament als 2,5V. Per calcular el valor eficaç i màxim es fa de la mateixa manera que en els casos anteriors (numsenyal=0 i numsenyal=1). Com que anteriorment s'ha obtingut el valor de tensió i ara ja es té el valor de intensitat, només cal multiplicar aquest 2 valors i ja s'obté la demanda de potència, la qual es dividida per 1000 per obtenir el resultat en KW. Per últim només queda torna a canviar l'entrada del ADC (P1.2 a P1.0) per tornar a veure un nou valor de tensió d'entrada després canviar (P1.0 o P1.1 a P1.2), per obtenir la intensitat i consegüentment obtenir un nou valor de demanda de potència.

```

    numsenyal=0;
    acvat=0;
    potencia=(tensio*vefintensi)/1000;

    ADC_Init();
  }
}
}

```

Com que la resolució de la placa ToolStick és de 10 bits y com a màxim se li pot introduir un valor analògic de 3,40 V, s'obtenen 3,30 mV/LSB. La sensibilitat del sensor és de 15 mV per amper, per tant com a mínim es pot detectar 0,22 A, en definitiva detectar aparells a partir de 50 vats aproximadament. A la pràctica ajustar aquest valor mínim resultat bastant complicat ja que aquest valor depèn si al sensor li esta arribant els 5 V d'alimentació estables. L'ajust mínim que s'ha pogut establir és de 500 vats cap endavant .

Per explicar una parts de l'última funció s'ha de tenir present el codi ASCII que és un codi de representació de caràcters que normalment utilitza una escala decimal del 0 al 127. També es pot utilitzar l'escala hexadecimal, octal o html . En aquest cas l'escala que s'utilitzarà es la decimal.

Aquests nombres decimals són transformats per la computadora en nombres binaris per així més endavant arribar a ser processats. Per tant, cada lletra que s'escriuigui correspondrà a cada un d'aquests codis.

Dec	Hx	Char	Dec	Hx	Html	Chr	Dec	Hx	Html	Chr	Dec	Hx	Html	Chr
0	0	<b>NUL</b> (null)	32	20	0x32	<b>Space</b>	64	40	0x40	<b>@</b>	96	60	0x96	<b>`</b>
1	1	<b>SOH</b> (start of heading)	33	21	0x33	<b>!</b>	65	41	0x41	<b>A</b>	97	61	0x97	<b>a</b>
2	2	<b>STX</b> (start of text)	34	22	0x34	<b>"</b>	66	42	0x42	<b>B</b>	98	62	0x98	<b>b</b>
3	3	<b>ETX</b> (end of text)	35	23	0x35	<b>#</b>	67	43	0x43	<b>C</b>	99	63	0x99	<b>c</b>
4	4	<b>EOT</b> (end of transmission)	36	24	0x36	<b>\$</b>	68	44	0x44	<b>D</b>	100	64	0x100	<b>d</b>
5	5	<b>ENQ</b> (enquiry)	37	25	0x37	<b>%</b>	69	45	0x45	<b>E</b>	101	65	0x101	<b>e</b>
6	6	<b>ACK</b> (acknowledge)	38	26	0x38	<b>&amp;</b>	70	46	0x46	<b>F</b>	102	66	0x102	<b>f</b>
7	7	<b>BEL</b> (bell)	39	27	0x39	<b>'</b>	71	47	0x47	<b>G</b>	103	67	0x103	<b>g</b>
8	8	<b>BS</b> (backspace)	40	28	0x40	<b>(</b>	72	48	0x48	<b>H</b>	104	68	0x104	<b>h</b>
9	9	<b>TAB</b> (horizontal tab)	41	29	0x41	<b>)</b>	73	49	0x49	<b>I</b>	105	69	0x105	<b>i</b>
10	A	<b>LF</b> (NL line feed, new line)	42	2A	0x42	<b>*</b>	74	4A	0x4A	<b>J</b>	106	6A	0x106	<b>j</b>
11	B	<b>VT</b> (vertical tab)	43	2B	0x43	<b>+</b>	75	4B	0x4B	<b>K</b>	107	6B	0x107	<b>k</b>
12	C	<b>FF</b> (NP form feed, new page)	44	2C	0x44	<b>,</b>	76	4C	0x4C	<b>L</b>	108	6C	0x108	<b>l</b>
13	D	<b>CR</b> (carriage return)	45	2D	0x45	<b>-</b>	77	4D	0x4D	<b>M</b>	109	6D	0x109	<b>m</b>
14	E	<b>SO</b> (shift out)	46	2E	0x46	<b>.</b>	78	4E	0x4E	<b>N</b>	110	6E	0x110	<b>n</b>
15	F	<b>SI</b> (shift in)	47	2F	0x47	<b>/</b>	79	4F	0x4F	<b>O</b>	111	6F	0x111	<b>o</b>
16	10	<b>DLE</b> (data link escape)	48	30	0x48	<b>0</b>	80	50	0x50	<b>P</b>	112	70	0x112	<b>p</b>
17	11	<b>DC1</b> (device control 1)	49	31	0x49	<b>1</b>	81	51	0x51	<b>Q</b>	113	71	0x113	<b>q</b>
18	12	<b>DC2</b> (device control 2)	50	32	0x50	<b>2</b>	82	52	0x52	<b>R</b>	114	72	0x114	<b>r</b>
19	13	<b>DC3</b> (device control 3)	51	33	0x51	<b>3</b>	83	53	0x53	<b>S</b>	115	73	0x115	<b>s</b>
20	14	<b>DC4</b> (device control 4)	52	34	0x52	<b>4</b>	84	54	0x54	<b>T</b>	116	74	0x116	<b>t</b>
21	15	<b>NAK</b> (negative acknowledge)	53	35	0x53	<b>5</b>	85	55	0x55	<b>U</b>	117	75	0x117	<b>u</b>
22	16	<b>SYN</b> (synchronous idle)	54	36	0x54	<b>6</b>	86	56	0x56	<b>V</b>	118	76	0x118	<b>v</b>
23	17	<b>ETB</b> (end of trans. block)	55	37	0x55	<b>7</b>	87	57	0x57	<b>W</b>	119	77	0x119	<b>w</b>
24	18	<b>CAN</b> (cancel)	56	38	0x56	<b>8</b>	88	58	0x58	<b>X</b>	120	78	0x120	<b>x</b>
25	19	<b>EM</b> (end of medium)	57	39	0x57	<b>9</b>	89	59	0x59	<b>Y</b>	121	79	0x121	<b>y</b>
26	1A	<b>SUB</b> (substitute)	58	3A	0x58	<b>:</b>	90	5A	0x5A	<b>Z</b>	122	7A	0x122	<b>z</b>
27	1B	<b>ESC</b> (escape)	59	3B	0x59	<b>;</b>	91	5B	0x5B	<b>[</b>	123	7B	0x123	<b>{</b>
28	1C	<b>FS</b> (file separator)	60	3C	0x60	<b>&lt;</b>	92	5C	0x5C	<b>\</b>	124	7C	0x124	<b> </b>
29	1D	<b>GS</b> (group separator)	61	3D	0x61	<b>=</b>	93	5D	0x5D	<b>]</b>	125	7D	0x125	<b>}</b>
30	1E	<b>RS</b> (record separator)	62	3E	0x62	<b>&gt;</b>	94	5E	0x5E	<b>^</b>	126	7E	0x126	<b>~</b>
31	1F	<b>US</b> (unit separator)	63	3F	0x63	<b>?</b>	95	5F	0x5F	<b>_</b>	127	7F	0x127	<b>DEL</b>

Il·lustració 41 – Codi ASCII

```

void main ( void )
{
    PCA_Init();
    ADC_Init();
    Port_IO_Init();
    Oscillator_Init();
    Interrupts_Init();
    UART0_Init();
    vegades8=0;
    bloqueigtarif=0;
    rebretarif=1;
    bloqueigpotencia=1;
    rebrepotencia=1;
    bloqueireresult=1;

```

La funció **void main** és la funció principal del programa, la qual permet l'execució de totes les funcions que s'han definit, així com inicialitzar a zero les principals variables, tals com vegades8 que es l'encarregada de comptabilitzar el nombre de mostres o el nombre de rectangles en un període. També hi ha variables de control pel flux d'informació com bloqueigtarif o rebretarif, que permeten tant l'enviament com la recepció d'aquesta.

```

while(1)
{
    if (bloqueigtarif==0)
    {
        if(TI0==1)
        {
            switch (index)
            {
                case 0:
                    TI0=0;
                    SBUF0=84; → T
                    index++;
                    break;
                case 1:
                    TI0=0;
                    SBUF0=97; → a
                    index++;
                    break;
                case 2:
                    TI0=0;
                    SBUF0=114; → r
                    index++;
                    break;
                case 3:
                    TI0=0;
                    SBUF0=105; → i
                    index++;
                    break;
                case 4:
                    TI0=0;
                    SBUF0=102; → f
                    index++;
                    break;
            }

```

Com que es necessita que el programa s'executi de manera cíclica es posa una condició que sempre es compleixi en aquest cas while(1). La transició bidireccional d'informació (enviament o recepció de informació) es podria fer mitjançant interrupcions però com que es vol adquirir mostres de senyals amb un marge de temps molt petit (temps crític), es donarà prioritat als càlculs mentre que la informació s'enviarà o es rebrà quan es pugui, per exemple en un interval de temps en el que no s'estiguin fent càlculs ni adquisicions. Per enviar o rebre informació s'ha de tenir en compte com funciona la UART o port sèrie. Consta d'un buffer de recepció i un buffer de transmissió, la disponibilitat d'aquests es basa en 2 bits de control TI0 que serveix per habilitar la transmissió de dades i RI0 que serveix per habilitar la recepció de dades. Els buffers poden guardar només un caràcter, per tant no es poden enviar cadenes de caràcters de cop. Primer s'envia la cadena de caràcters " Tarifa (Eur/KWh) :", que està controlada per la variable bloqueigtarif.

```

case 5:
    TI0=0;
    SBUF0=97; → a
    index++;
    break;
case 6:
    TI0=0;
    SBUF0=32; → Espai
    index++;
    break;
case 7:
    TI0=0;
    SBUF0=40; → (
    index++;
    break;
case 8:
    TI0=0;
    SBUF0=69; → E
    index++;
    break;
case 9:
    TI0=0;
    SBUF0=117; → u
    index++;
    break;
case 10:
    TI0=0;
    SBUF0=114; → r
    index++;
    break;
case 11:
    TI0=0;
    SBUF0=47; → /
    index++;
    break;
case 12:
    TI0=0;
    SBUF0=75; → K
    index++;
    break;
case 13:
    TI0=0;
    SBUF0=87; → W
    index++;
    break;

```

Com que a la funció **void UART0\_Init** s'ha establert TI0=1 es podrà començar a enviar el primer caràcter. Per saber quin caràcter s'ha d'enviar posteriorment s'utilitza un estructura switch, que a partir de l'increment de la variable index (que està associada a case) permet entrar en els diferents casos i així poder enviar cada caràcter. Per introduir correctament els caràcters que es vulgui al buffer de transmissió s'utilitzarà el codi ASCII, el qual s'ha enunciat anteriorment. Quan s'entra a cada cas el bit TI0 es posa manualment a zero i quan el caràcter ja s'ha enviat a la pantalla de caràcters transmesos el bit TI0 es posa a u automàticament per indicar que està preparat per enviar un nou caràcter. Això es produeix per cada un dels casos fins que s'arriba al últim cas, en que ja no s'incrementa la variable index sino que es posa a zero, es posa a u la variable bloqueigtarif perquè no es torni a enviar la cadena " Tarifa (Eur/KWh) : " fins que li pertoqui i a més es posa a zero la variable rebretarif per permetre que l'usuari introdueixi el preu per KWh que vulgui.

case 14:

```
TI0=0;
SBUF0=104; → h
index++;
break;
```

case 15:

```
TI0=0;
SBUF0=41; → )
index++;
break;
```

case 16:

```
TI0=0;
SBUF0=58; → :
index=0;
bloqueigtarif=1;
rebretarif=0;
break;
}
```

S'agafa cada caràcter del buffer de recepció i es posa en el vector vectorif.

```
}
if(rebretarif==0)
{
  if (RI0==1)
  {
    RI0=0;
    switch (contador)
    {
```

Després s'agafa el dígit del vector i es posa en el buffer de transmissió.

case 0:

```
vectorif[0]=SBUF0;
SBUF0=vectorif[0]; → t0
contador++;
break;
```

case 1:

```
vectorif[1]=SBUF0;
SBUF0=vectorif[1]; → t1
contador++;
break;
```

case 2:

```
vectorif[2]=SBUF0;
SBUF0=vectorif[2]; → t2
contador++;
break;
```

case 3:

```
vectorif[3]=SBUF0;
SBUF0=vectorif[3]; → t3
contador++;
break;
```

Quan la variable rebretarif està a zero vol dir que l'usuari pot introduir el preu de la tarifa, en aquest cas per rebre el valor que proporcionarà l'usuari s'utilitzarà el bit RI0. Mentre l'usuari no teclegi cap dígit RI0 serà zero, però en el moment que teclegi RI0 es posarà automàticament a u i podrà consegüentment entrar en cada un dels casos, tenint en compte que en aquest cas la variable associada a cada cas és contador. Quan s'arriba al últim cas es posa a zero la variable t i contador. El valor que ha introduït l'usuari es guarda en un vector i posteriorment es transforma a decimal a partir de la funció atof, inclosa en la llibreria stdlib.h.



```

case 4:
    t=0;
    contador=0;
    tarifa = atof (vectarif);
    rebretarif=1;
    bloqueigpotencia=0;
    break;
}
}
if (bloqueigpotencia==0)
{
    if(TI0==1)
    {
        switch (rep)
        {
            case 0:
                TI0=0;
                SBUF0=10; → Salt línia
                rep++;
                break;
            case 1:
                TI0=0;
                SBUF0=68; → D
                rep++;
                break;
            case 2:
                TI0=0;
                SBUF0=101; → e
                rep++;
                break;
            case 3:
                TI0=0;
                SBUF0=109; → m
                rep++;
                break;
            case 4:
                TI0=0;
                SBUF0=97; → a
                rep++;
                break;
            case 5:
                TI0=0;
                SBUF0=110; → n
                rep++;
                break;
        }
    }
}

```

Per últim només queda posar a u rebretarif perquè l'usuari no pugui introduir més informació i es posa a zero bloqueigpotencia per permetre l'enviament de la cadena "Demanda maxima potència (KW):".

Arribats a aquest punt es torna a treballar amb el bit TI0 i l'estructura switch a on la variable rep va associada a cada cas de la estructura. El procés és igual que quan s'ha enviat la cadena "Tarifa (Eur/KWh) :", quan s'arriba a cada cas es posa a zero el bit TI0 es guarda el caràcter al buffer de transmissió i es va incrementant la variable rep per poder entrar als altres casos i així poder enviar els caràcters restants. I a l'últim cas es posa a 1 bloqueigpotencia, perquè la cadena "Demanda maxima potència (KW):" no es torni a mostra fins que no li pertoqui, és a dir, quan es pari i es torni a executa el programa. També es posa a zero rebrepotencia per permetre que l'usuari introdueixi la demanda de potència màxima que vulgui.

case 6:

```
TI0=0;  
SBUF0=100; → d  
rep++;  
break;
```

case 7:

```
TI0=0;  
SBUF0=97; → a  
rep++;  
break;
```

case 8:

```
TI0=0;  
SBUF0=32; → Espai  
rep++;  
break;
```

case 9:

```
TI0=0;  
SBUF0=109; → M  
rep++;  
break;
```

case 10:

```
TI0=0;  
SBUF0=97; → a  
rep++;  
break;
```

case 11:

```
TI0=0;  
SBUF0=120; → x  
rep++;  
break;
```

case 12:

```
TI0=0;  
SBUF0=105; → i  
rep++;  
break;
```

case 13:

```
TI0=0;  
SBUF0=109; → m  
rep++;  
break;
```

case 14:

```
TI0=0;  
SBUF0=97; → a  
rep++;  
break;
```



case 15:

TI0=0;

SBUF0=32; → Espai

rep++;

break;

case 16:

TI0=0;

SBUF0=112; → p

rep++;

break;

case 17:

TI0=0;

SBUF0=111; → o

rep++;

break;

case 18:

TI0=0;

SBUF0=116; → t

rep++;

break;

case 19:

TI0=0;

SBUF0=101; → e

rep++;

break;

case 20:

TI0=0;

SBUF0=110; → n

rep++;

break;

case 21:

TI0=0;

SBUF0=99; → c

rep++;

break;

case 22:

TI0=0;

SBUF0=105; → i

rep++;

break;

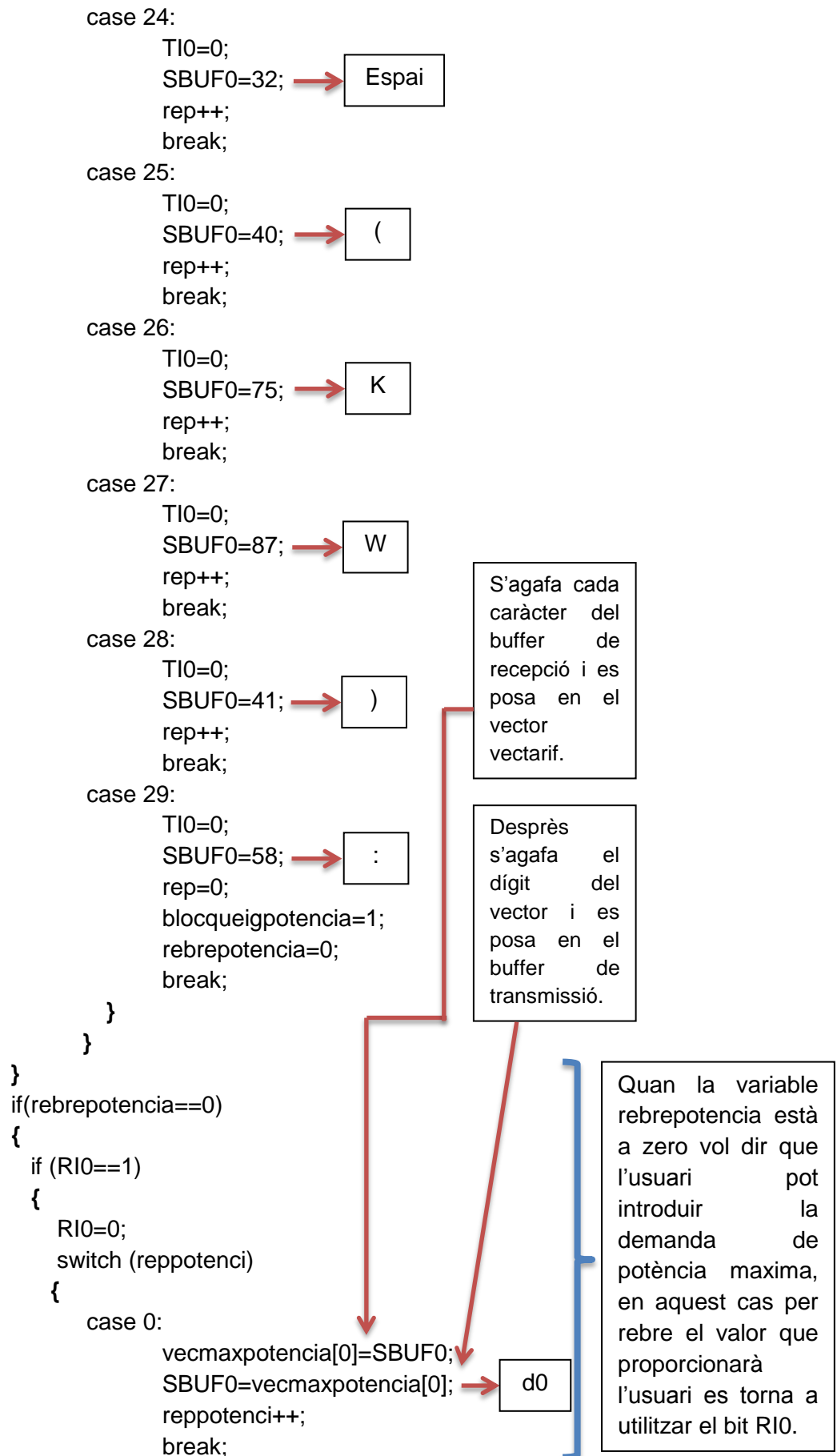
case 23:

TI0=0;

SBUF0=97; → a

rep++;

break;



```

case 1:
    vecmaxpotencia[1]=SBUF0;
    SBUF0=vecmaxpotencia[1]; → d1
    reppotenci++;
    break;
case 2:
    vecmaxpotencia[2]=SBUF0; → d2
    SBUF0=vecmaxpotencia[2];
    reppotenci++;
    break;
case 3:
    vecmaxpotencia[3]=SBUF0;
    SBUF0=vecmaxpotencia[3]; → d3
    reppotenci++;
    break;
case 4:
    vecmaxpotencia[4]=SBUF0;
    SBUF0=vecmaxpotencia[4]; → d4
    reppotenci++;
    break;
case 5:
    reppotenci=0;
    bloqueigpotencia=1;
    t=0;
    contador=0;
    SBUF0=10; → Salt línia
    potencimax = atof (vecmaxpotencia);
    frecuencia=50;
    periode= 1/frecuencia;
    periodemostres=periode/200;
    sEOS_Init_Timer2(periodemostres);
    break;
    }
    }
}
if (bloqueireresult==0)
{
    if(TI0==1)
    {
        switch (mosampli)
        {
            case 0:
                TI0=0;
                SBUF0=68; → D
                mosampli++;
                break;

```

Metre l'usuari no teclegi cap dígit RI0 serà zero, però en el moment que teclegi RI0 es posarà automàticament a u i podrà conseqüentment entrar en cada un del casos, tenint en compte que en aquest cas la variable associada a cada cas és reppotenci. Quan s'arriba al últim cas es posa a zero la variable t , reppotenci i contador. El valor que ha introduït l'usuari es guarda en el vector vecmaxpotencia i posteriorment es transforma a decimal a partir de la funció atof. Després es posa a u bloqueigpotencia perquè l'usuari no introdueixi més informació, ja s'ha agafat la necessària i s'estableixen els paràmetres per començar a fer l'adquisició de les mostres i els càlculs, com per exemple la freqüència, període o configuració del timer 2 perquè provoqui una interrupció quan es vulgui adquirir una mostra.

La variable bloqueireresult es posarà a zero una vegada el tenen tots el càlculs i esta tot a punt per poder enviar la informació cap al PC (pàg 25). En aquest punt es torna a treballar amb el bit TI0 i la estructura switch. Ara la variable associada a cada cas és mosampli.

case 1:

```
TI0=0;  
SBUF0=101; → e  
mosampli++;  
break;
```

case 2:

```
TI0=0;  
SBUF0=109; → m  
mosampli++;  
break;
```

case 3:

```
TI0=0;  
SBUF0=97; → a  
mosampli++;  
break;
```

case 4:

```
TI0=0;  
SBUF0=110; → n  
mosampli++;  
break;
```

case 5:

```
TI0=0;  
SBUF0=100; → d  
mosampli++;  
break;
```

case 6:

```
TI0=0;  
SBUF0=97; → a  
mosampli++;  
break;
```

case 7:

```
TI0=0;  
SBUF0=32; → Espai  
mosampli++;  
break;
```

case 8:

```
TI0=0;  
SBUF0=77; → M  
mosampli++;  
break;
```

case 9:

```
TI0=0;  
SBUF0=97; → a  
mosampli++;  
break;
```

case 10:

```
TI0=0;  
SBUF0=120;  
mosampli++;  
break;
```

x

case 11:

```
TI0=0;  
SBUF0=32;  
mosampli++;  
break;
```

Espai

case 12:

```
TI0=0;  
SBUF0=vecmaxpotencia[0];  
mosampli++;  
break;
```

p0

case 13:

```
TI0=0;  
SBUF0=vecmaxpotencia[1];  
mosampli++;  
break;
```

p1

case 14:

```
TI0=0;  
SBUF0=vecmaxpotencia[2];  
mosampli++;  
break;
```

p2

case 15:

```
TI0=0;  
SBUF0=vecmaxpotencia[3];  
mosampli++;  
break;
```

p3

case 16:

```
TI0=0;  
SBUF0=vecmaxpotencia[4];  
mosampli++;  
break;
```

p4

case 17:

```
TI0=0;  
SBUF0=32;  
mosampli++;  
break;
```

Espai

case 18:

```
TI0=0;  
SBUF0=75;  
mosampli++;  
break;
```

K

case 19:

```
TI0=0;  
SBUF0=87; → W  
mosampli++;  
break;
```

case 20:

```
TI0=0;  
SBUF0=44; → ,  
mosampli++;  
break;
```

case 21:

```
TI0=0;  
SBUF0=32; → Espai  
mosampli++;  
break;
```

case 22:

```
TI0=0;  
SBUF0=68; → D  
mosampli++;  
break;
```

case 23:

```
TI0=0;  
SBUF0=101; → e  
mosampli++;  
break;
```

case 24:

```
TI0=0;  
SBUF0=109; → m  
mosampli++;  
break;
```

case 25:

```
TI0=0;  
SBUF0=97; → a  
mosampli++;  
break;
```

case 26:

```
TI0=0;  
SBUF0=110; → n  
mosampli++;  
break;
```

case 27:

```
TI0=0;  
SBUF0=100; → d  
mosampli++;  
break;
```

case 28:

```
TI0=0;  
SBUF0=97; →  
mosampli++;  
break;
```

a

case 29:

```
TI0=0;  
SBUF0=32; →  
mosampli++;  
break;
```

Espai

case 30:

```
TI0=0;  
SBUF0=vecpotencia[0]; →  
mosampli++;  
break;
```

da0

case 31:

```
TI0=0;  
SBUF0=vecpotencia[1]; →  
mosampli++;  
break;
```

da1

case 32:

```
TI0=0;  
SBUF0=vecpotencia[2]; →  
mosampli++;  
break;
```

da2

case 33:

```
TI0=0;  
SBUF0=vecpotencia[3]; →  
mosampli++;  
break;
```

da3

case 34:

```
TI0=0;  
SBUF0=vecpotencia[4]; →  
mosampli++;  
break;
```

da4

case 35:

```
TI0=0;  
SBUF0=32; →  
mosampli++;  
break;
```

Espai

case 36:

```
TI0=0;  
SBUF0=75; →  
mosampli++;  
break;
```

K

case 37:

```
TI0=0;  
SBUF0=87; → W  
mosampli++;  
break;
```

case 38:

```
TI0=0;  
SBUF0=44; → ,  
mosampli++;  
break;
```

case 39:

```
TI0=0;  
SBUF0=32; → Espai  
mosampli++;  
break;
```

case 40:

```
TI0=0;  
SBUF0=69; → E  
mosampli++;  
break;
```

case 41:

```
TI0=0;  
SBUF0=110; → n  
mosampli++;  
break;
```

case 42:

```
TI0=0;  
SBUF0=101; → e  
mosampli++;  
break;
```

case 43:

```
TI0=0;  
SBUF0=114; → r  
mosampli++;  
break;
```

case 44:

```
TI0=0;  
SBUF0=103; → g  
mosampli++;  
break;
```

case 45:

```
TI0=0;  
SBUF0=105; → i  
mosampli++;  
break;
```



case 46:

```
TI0=0;  
SBUF0=97; →  
mosampli++;  
break;
```

a

case 47:

```
TI0=0;  
SBUF0=32; →  
mosampli++;  
break;
```

Espai

case 48:

```
TI0=0;  
SBUF0=vecenergia[0]; →  
mosampli++;  
break;
```

e0

case 49:

```
TI0=0;  
SBUF0=vecenergia[1]; →  
mosampli++;  
break;
```

e1

case 50:

```
TI0=0;  
SBUF0=vecenergia[2]; →  
mosampli++;  
break;
```

e2

case 51:

```
TI0=0;  
SBUF0=vecenergia[3]; →  
mosampli++;  
break;
```

e3

case 52:

```
TI0=0;  
SBUF0=vecenergia[4]; →  
mosampli++;  
break;
```

e4

case 53:

```
TI0=0;  
SBUF0=vecenergia[5]; →  
mosampli++;  
break;
```

e5

case 54:

```
TI0=0;  
SBUF0=32; →  
mosampli++;  
break;
```

Espai

case 55:

```
TI0=0;  
SBUF0=75;  
mosampli++;  
break;
```

→ K

case 56:

```
TI0=0;  
SBUF0=87;  
mosampli++;  
break;
```

→ W

case 57:

```
TI0=0;  
SBUF0=104;  
mosampli++;  
break;
```

→ h

case 58:

```
TI0=0;  
SBUF0=44;  
mosampli++;  
break;
```

→ ,

case 59:

```
TI0=0;  
SBUF0=32;  
mosampli++;  
break;
```

→ Espai

case 60:

```
TI0=0;  
SBUF0=80;  
mosampli++;  
break;
```

→ P

case 61:

```
TI0=0;  
SBUF0=114;  
mosampli++;  
break;
```

→ r

case 62:

```
TI0=0;  
SBUF0=101;  
mosampli++;  
break;
```

→ e

case 63:

```
TI0=0;  
SBUF0=117;  
mosampli++;  
break;
```

→ u

case 64:  
 TI0=0;  
 SBUF0=32; → Espai  
 mosampli++;  
 break;

case 65:  
 TI0=0;  
 SBUF0=vecpreu[0]; → p0  
 mosampli++;  
 break;

case 66:  
 TI0=0;  
 SBUF0=vecpreu[1]; → p1  
 mosampli++;  
 break;

case 67:  
 TI0=0;  
 SBUF0=vecpreu[2]; → p2  
 mosampli++;  
 break;

case 68:  
 TI0=0;  
 SBUF0=vecpreu[3]; → p3  
 mosampli++;  
 break;

case 69:  
 TI0=0;  
 SBUF0=32; → Espai  
 mosampli++;  
 break;

case 70:  
 TI0=0;  
 SBUF0=69; → E  
 mosampli++;  
 break;

case 71:  
 TI0=0;  
 SBUF0=117; → u  
 mosampli++;  
 break;

case 72:  
 TI0=0;  
 SBUF0=114; → r  
 mosampli++;  
 break;

```

case 73:
    TI0=0;
    mosampli=0;
    SBUF0=10; → Salt línia
    bloqueireult=1;
    break;
}
}
}
}
}

```


Per últim es posa a zero la variable mosampli perquè quan es tinguin càlculs nous pugui entrar un altre cop a tots els casos i així mostrar la informació correctament. També es posa a u la variable bloqueireult perquè no es torni a mostrar la informació fins que pertoqui, és a dir, quan es tornin a fer càlculs nous.

### 5.1.2 Simulació del funcionament del codi amb el software µVision IDE

Per comprovar el correcte funcionament del codi és farà una simulació amb el programa µVision IDE. Per poder fer aquesta simulació es necessiten generar de manera virtual, els senyals que s'introduiran al microcontrolador Intel 8051F330. Aquests senyals són:

- Senyal de tensió d'entrada corresponent a l'escala de 0 – 400 volts, que anirà al port P1.0 del microcontrolador.
- Senyal de tensió d'entrada corresponent a l'escala de 0 – 60 volts, que anirà al port P1.1 del microcontrolador.
- Senyal de tensió de sortida del sensor de corrent, que anirà al port P1.2 del microcontrolador.

Per generar aquest senyals es fa servir un projecte de µVision sobre el qual s'afegeix el codi del monitor d'energia. Per modificar la amplitud i el nivell de tensió continua dels senyals virtuals es modifica un arxiu d'extensió text que es troba ubicat en el mateix projecte. A continuació s'enuncien els passos que s'han seguit per fer la simulació.

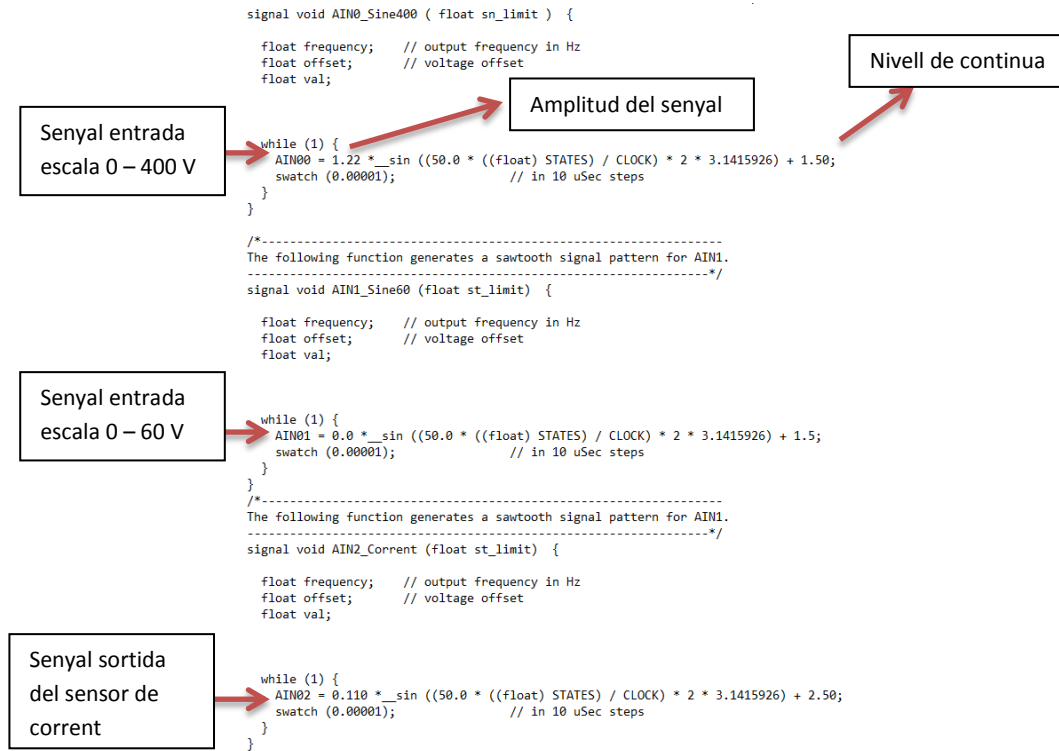
- 1- Una vegada s'ha escrit el codi del monitor d'energia es clica la icona Rebuild  per comprovar que no hi cap error en el codi. Si el codi no presenta cap problema el programa ha de mostrar que hi ha 0 errors i 0 warnings.

```
Rebuild target 'C8051F0330'  
assembling STARTUP.A51...  
compiling main.C...  
linking...  
Program Size: data=18.6 xdata=195 code=7266  
"ADC" - 0 Error(s), 0 Warning(s).
```




Il·lustració 42 – Missatge del programa µVision per indicar que el codi no presenta problemes

- 2- Modificar l'amplitud i el nivell de continua de cada una de les 3 senyals, modificant l'arxiu debug.ini que es troba dins el projecte. Per aquest exemple es suposarà que es vol mesurar el consum d'un aparell de 1200 wats, tenint en compte una tensió d'entrada de 230 V AC. Es calcula la intensitat que hauria de detectar el sensor de corrent.  $\text{Intensitat} = \text{Potència} / \text{voltatge} = 1200 / 230 = 5,21 \text{ A}$ . Com s'ha parlat a l'apartat 4, es volia reduir l'escala de 0 – 400 volts (que va referit al valor màxim del senyal que pot detectar el monitor) a una escala de 0 – 1,5 volts que ve juntament amb un nivell de continua de 1,5 V (el rang de tensió d'entrada que admet el microcontrolador s'ha fixat entre 0 – 3 volts ). Per un senyal de 230 V AC, el seu valor màxim és 325 volts, que correspondrà a 1,22 volts en l'escala de 0 – 1,5 volts. Però com que aquesta escala va juntament amb un nivell de continua de 1,5 volts, el valor que haurà de llegir el microcontrolador serà de 1,5 + 1,22 volts, és a dir, 2,77 volts.  
A l'apartat 4.2 es va parlar que el sensor de corrent podia detectar un rang de corrent des de -100 fins a 100 A (tensió de sortida del sensor de -1,5 volts per -100 A i 1,5 volts per 100A ), també es va veure que la seva tensió de referència era de 2,5 volts, per tant si el sensor detecta -100 A la tensió de sortida del

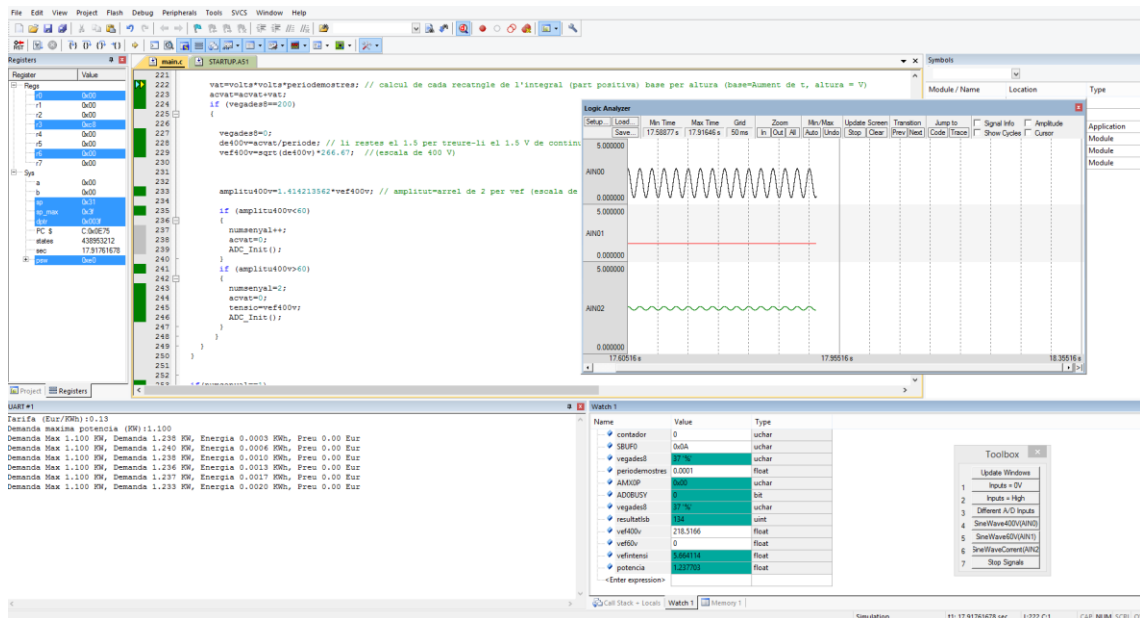
sensor serà de  $2,5 - 1,5 = 1$  volt i si detecta 100 A la tensió de sortida del sensor serà de  $2,5 + 1,5 = 4$  volts. Com que s'ha establert que el rang d'entrada que pot detectar el microcontrolador és de 0 – 3 volts, el rang de corrent que podrà detectar el monitor d'energia serà de -100 A fins a 75 A. Com la sensibilitat del sensor és de 15 mV/ A, la tensió sinusoidal de sortida del sensor per un corrent de 5,21 A (consum del aparell de 1200 wats) ha de ser de 2,50 volts de continua i 0.110 volts d'amplitud.



Il·lustració 43 – Definició dels senyals virtuals al arxiu debug.ini

- 3- Per començar la simulació es clica la icona Start  Debug.
- 4- Per engegar els senyals s'ha d'anar a la finestra anomenada Tollbox i clica els botons 4 SineWave400V, 5 SineWave60V i 6 SineWaveCorrent
- 5- Es clica la icona Analysis Windows  per veure la forma d'ona dels senyals. Per poder veure-les s'han d'arrossegar a aquesta finestra des de la pestanya Virtual Registers, sent les senyals AIN0 (SineWave400V), AIN1(SineWave60V) i AIN2 (SineWaveCorrent) .
- 6- Es clica la icona Run  per començar la simulació.

A continuació es mostrarà una captura de la simulació.




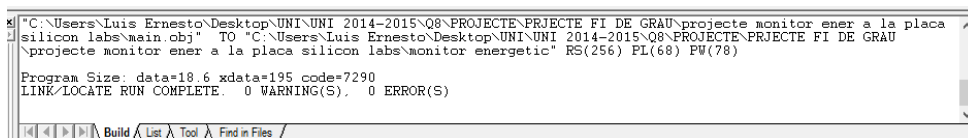
II-il·lustració 44 – Definició dels senyals virtuals al arxiu debug.ini

Com es pot observar a la il·lustració anterior a la simulació es va veient les formes d'ona de les senyals d'entrada, les variables corresponents on hi ha la tensió eficaç, corrent eficaç i la demanda potència, així com també els caràcters que s'envien per la UART.




### 5.1.3 Comprovació del funcionament del codi a la placa ToolStick F330 DC amb el software Silicon Labs IDE

Per comprovar que el codi funciona correctament a la realitat es fa servir el programa proporcionat pels fabricants de la placa ToolStick F330 DC, Silicon Labs IDE. S'ha de tenir present que per poder fer la comprovació s'ha de tenir el circuit analògic, muntat i alimentat correctament. A continuació s'enuncien els passos que s'han seguit per fer la comprovació.

- 1- S'ha de crear un projecte a on es van guardant tots els arxius, per fer-ho es clica la pestanya File → New → Project → Project Type → C source project → i es fica la ubicació on es vol guardar.
- 2- Es clica la icona Rebuild All  per comprovar que no hi ha cap problema amb el codi. Si el codi no presenta cap problema el programa ha de mostrar un missatge dient que hi ha 0 errors i 0 warnings.

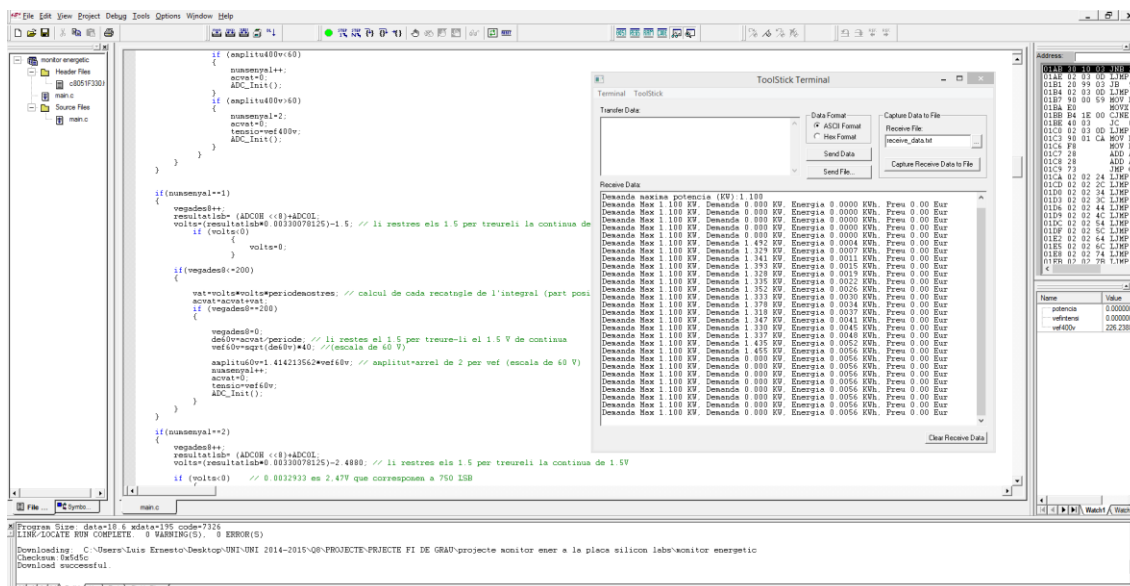


Il·lustració 45 – Missatge del programa Silicon Labs IDE per indicar que el codi no presenta problemes

- 3- Es clica la icona Connect  per connectar la placa amb l'ordinador.
- 4- Es clica la icona Download Code  per carregar el codi a la placa.
- 5- Es clica la pestanya Tools → ToolStick Terminal (l'eina que ajudar a l'enviament o la recepció de dades).
- 6- Es clica la pestanya ToolStick → Settings.. → i es posa el Baud Rate a 9600 → OK.
- 7- Es clica la icona Go  per començar a executar el programa.
- 8- S'introdueix primer la tarifa Eur/KWh i després s'introdueix la demana maxima de potència en KW.
- 9- S'engega l'aparell, que en aquest cas serà un assecador de cabells de 1200 vats o el que es equivalent 1,2 KW.

A continuació es mostrarà una captura de la comprovació del codi a la realitat. En aquest cas es mesurarà el consum d'una aspiradora de 1400 vats de potència.





**Il·lustració 46 – Comprovació del codi a la realitat utilitzant una aspiradora de 1,4 KW**

En la il·lustració anterior es pot observar que l'eina ToolStick Terminal va mostrant la informació en temps real, cada segon. Tal i com es va veure en el programa de simulació µVision a on es podien veure els valors de les variables, en el programa proporcionat pels fabricants de la placa, Silocon Labs IDE també es pot anar veient els valors de les variables que s'hagin arrossegat a la finestra watch, ubicada a la part de inferior a la dreta de la pantalla.

## 5.2 Programació de l'adquisició de dades amb el software LabVIEW

En aquesta apartat s'explicarà com s'ha realitzat el codi amb LabVIEW que permet rebre els caràcters que son enviats pel microcontrolador i separar el valors numèrics de les lletres per posteriorment enviar aquests valors numèrics a un gràfic. D'aquesta manera es pot veure l'historial del consum dels aparells que s'han utilitzat.

### 5.2.1 Explicació del codi

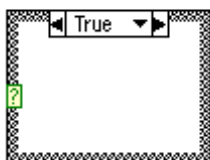
Per explicar el codi, primer s'ha de entendre les funcions que s'utilitzen en el codi, així doncs s'anirà enunciat cada funció amb el seu símbol i la seva funcionalitat.

#### While Loop



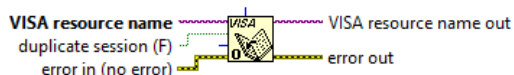
Estructura que executa un codi de manera cíclica, fins al moment en que l'usuari clica el botó STOP.

#### Case Structure



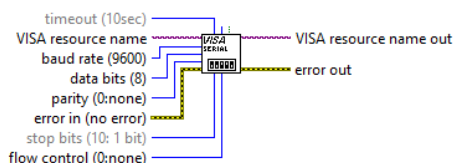
Estructura que executa un codi o altre ( cas True o cas False )en funció d'una condició booleana.

#### VISA Open



Funció que serveix obre sessió del mòdul VISA, amb el es pot llegir o enviar dades pel port sèrie.

#### VISA Configure Serial Port



Configuració del port sèrie en el que hi consten paràmetres com el nom del port que s'utilitzarà, el baud rate entre d'altres.

#### VISA Read

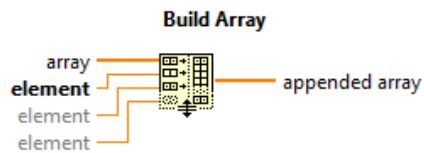


Mòdul de VISA que serveix per llegir caràcters del port sèrie.

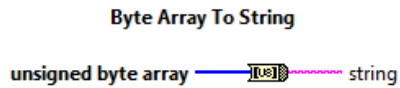
#### String To Byte Array



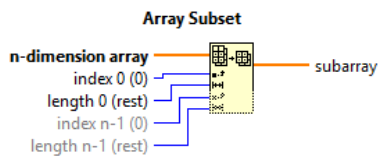
Funció que serveix per dividir una cadena de caràcters en caràcters individuals de tipus Byte.



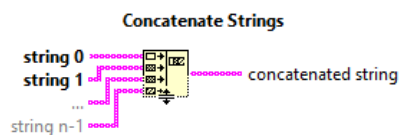
Funció que serveix per construir una matriu de n elements.



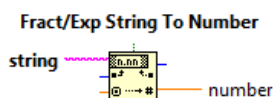
Funció que serveix per transformar una matriu de tipus byte en una tipus string.



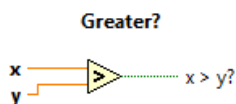
Funció que serveix per agafa una porció d'una matriu.



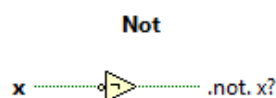
Funció que serveix per concatenar strings d'una porció d'una matriu.



Funció que serveix per transforma un string en un valor numèric.



Funció que serveix per comparar 2 valors, si x és més gran que y, retorna TRUE i en cas contrari retorna FALSE.



Funció que serveix per negar una variable de tipus booleà.



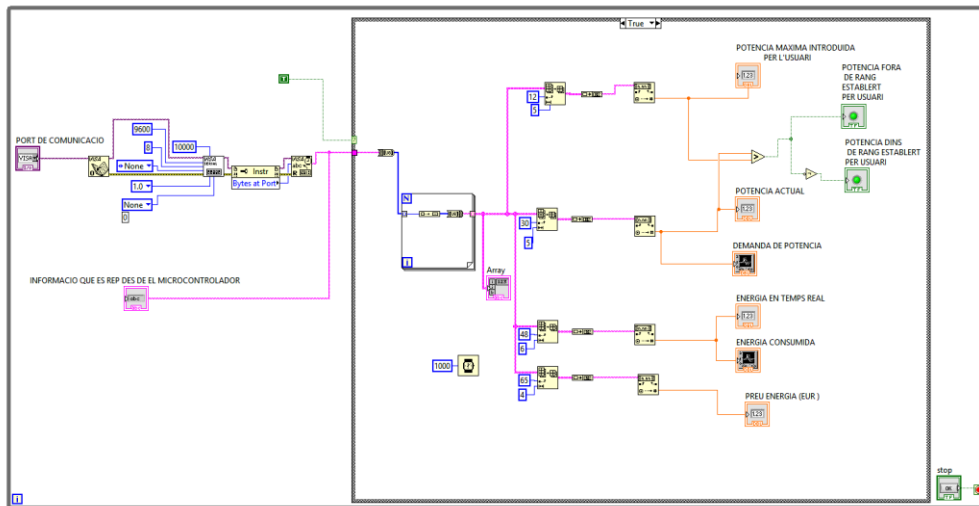
Element de tipus booleà que serveix com a indicador LED.



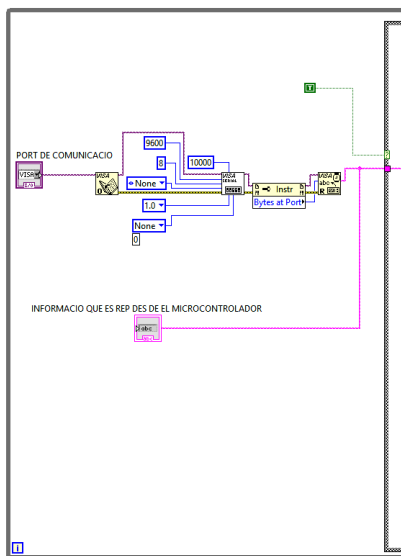
Element de tipus float que serveix per mostrar el valor d'una variable.



Element que serveix per gràfica valors numèrics, en temps real.

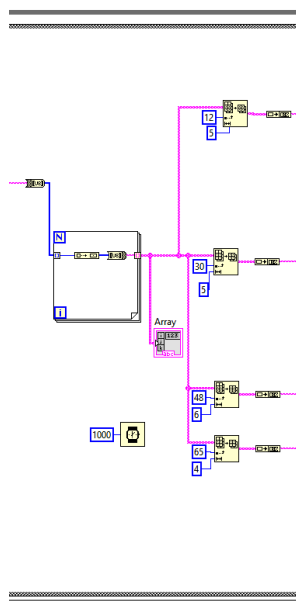


II-lustració 47 – Codi d'adquisició de dades complet



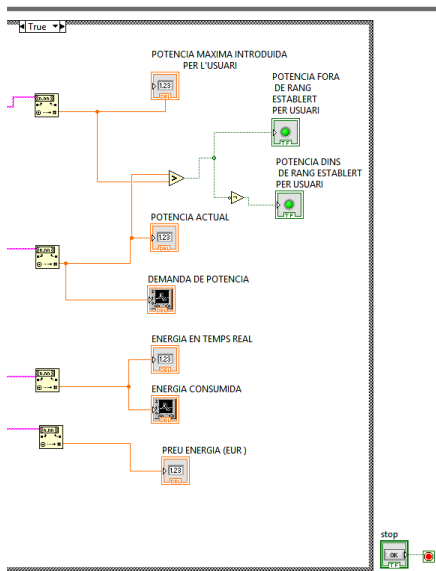
II-lustració 48 – Fragment 1 del codi

S'obre la sessió VISA, a la qual s'introdueix el port amb el que treballarà, en aquest cas es tracta del port COM1. Després es configura el port sèrie introduint el baud rate, el nombre de bits d'informació, posar o no bit de paritat i el timeout (si el programa no presenta cap canvi o transició, quan ha transcorregut el temps timeout, el programa s'atura). Després el mòdul VISA de lectura, mira si hi ha cap cadena de caràcters (de tipus string) en el buffer de recepció del port sèrie. Si hi ha caràcters el mòdul VISA els envia a l'estructura case, que sempre s'executa a través d'una condició TRUE permanent.



II-lustració 49 – Fragment 2 del codi

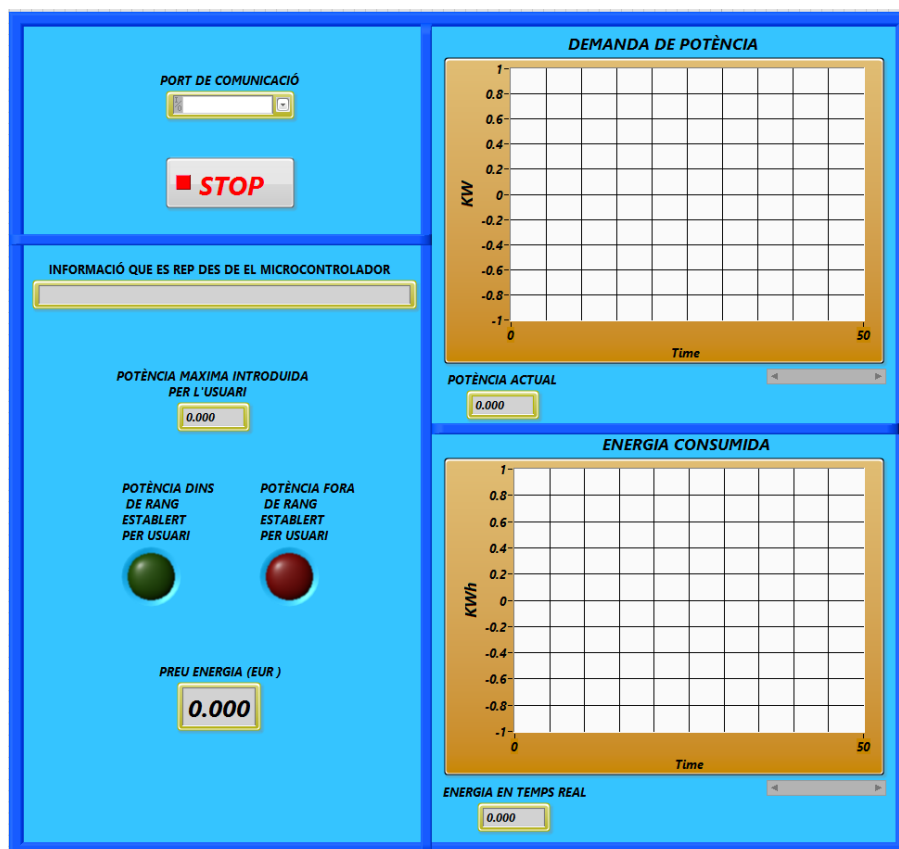
Quan arriba una cadena de caràcters tipus string es transformen en caràcters individuals de tipus Byte. Després es van emmagatzemant en una matriu de tipus Byte, que posteriorment es transforma en una matriu de tipus string. El fet de fer totes aquestes transformacions es deu a que en LabVIEW no es pot passar de una cadena de caràcters de tipus string a una matriu de tipus string. Un cop feta la transformació s'agafa la porció de la matriu que interessa, és a dir, els caràcters que siguin números, i es concatenen en una cadena de caràcters de tipus string.



Il·lustració 50 – Fragment 3 del codi

Un cop arriba la cadena de caràcters de tipus string on hi ha els números que interessin, es transforma en un numero de tipus float. Una vegada que es tenen les cadenes de caràcters de tipus string (que contenen el valor de la demanda de potència maxima establert per l'usuari, la demanda de potència actual, el valor de l'energia i el valor del preu de l'energia) i que han estat transformades en valors numèrics, es grafiquen els valors de la demanda de potència actual i energia, es compara el valor de potència introduït per l'usuari i el valor de la potència actual. Així com també es visualitza els valors del preu, energia consumida, valor de potència màxim establert per l'usuari i potència

Finalment es mostrarà el panell amb el que l'usuari podrà veure l'actualització del consum i de l'energia, cada segon, així com també dos indicadors lluminosos que indiquen si la demanda actual de potència sobrepassa o no el valor que ha introduït l'usuari en l'eina ToolStick Terminal.



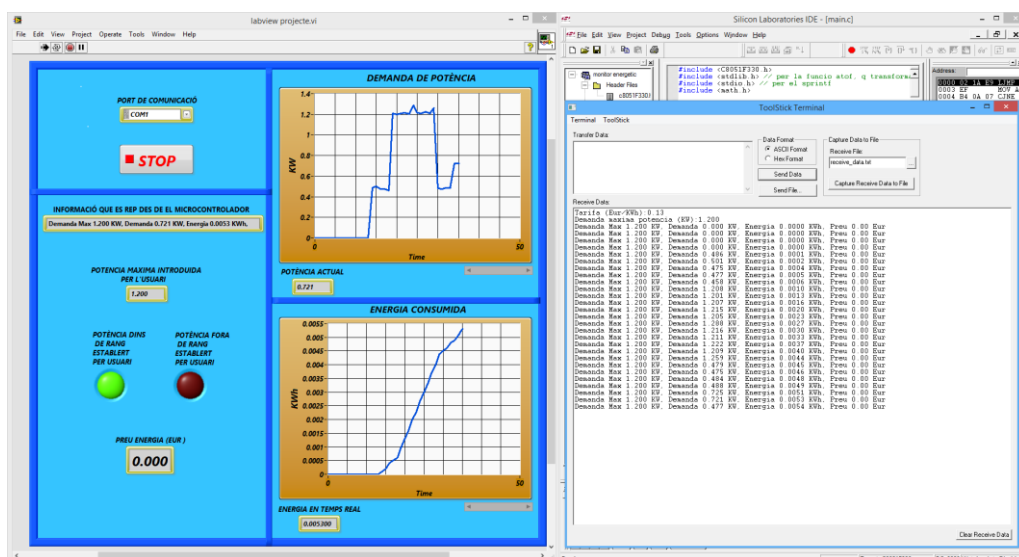
Il·lustració 51 – Panell que adquireix les dades que li envia el microcontrolador

## 5.2.2 Comprovació del funcionament del codi

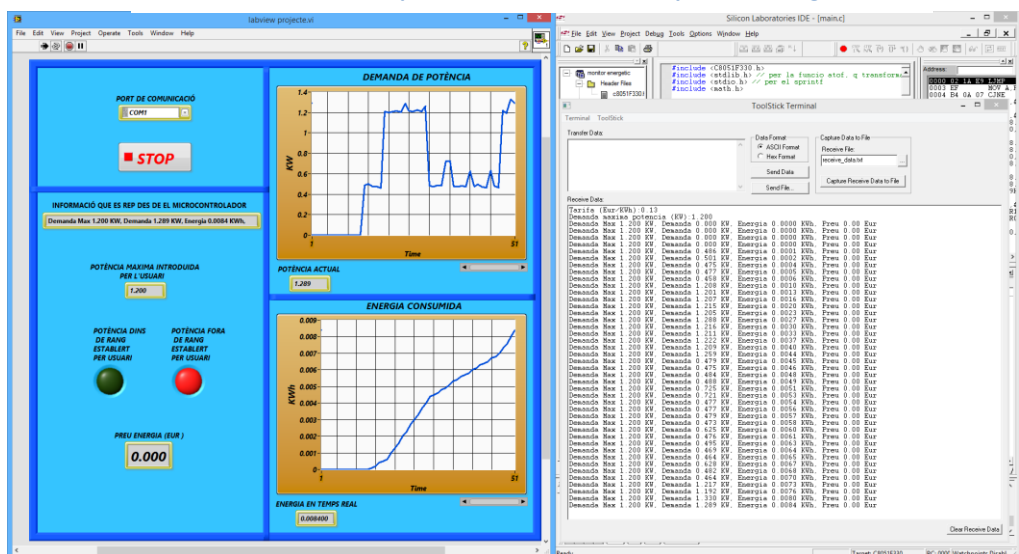
Per comprovar el funcionament del codi en LabVIEW, el circuit analògic ha d'estar muntat i alimentat correctament. També es necessita executar el codi de la placa ToolStick amb el programa Silicon Labs, ja que aquest és el que permetrà l'enviament dels caràcters des de el microcontrolador cap al ordinador. A continuació s'enuncien els passos que s'han seguit per adquirir les dades, un cop el microcontrolador estigui enviant caràcters.

- 1- A la finestra del panell on posa PORT DE COMUNICACIÓ es ficarà COM1.
- 2- Es clica la icona Run ➡ per començar l'adquisició de dades.
- 3- Quan es vulgui para l'adquisició de dades simplement es clica el botó STOP.

A continuació es mostrarà dos il·lustracions durant la comprovació del codi d'adquisició de dades. Per aquest comprovació s'ha utilitzat un assecador de cabells que té una potència màxima de 1200 wats i una mínima de 600 aproximadament.



Il·lustració 52 – Adquisició de dades durant els primers 35 segons



Il·lustració 53 – Adquisició de dades durant els primers 51 segons

## 6 Muntatge final

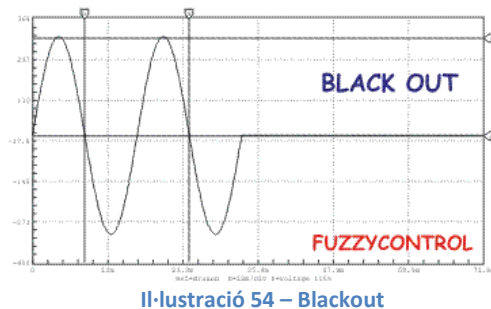
En aquest apartat s'expliquen les consideracions que s'han hagut de tenir en compte a l'hora de fer el muntatge del prototip del monitor d'energia així com també es volen mostrar una sèrie de proves del prototip, amb diferents aparells per comprovar que el monitor d'energia elèctrica funciona correctament.

### 6.1 Consideracions durant el muntatge final

Per realitzar el muntatge del prototip del monitor d'energia elèctrica s'han tingut presents les següents consideracions:

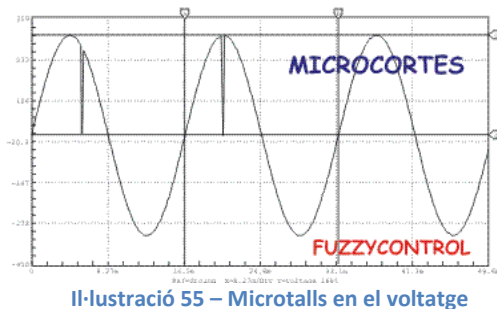
- Si es vol treballar amb un senyal d'entrada d'una instal·lació d'un habitatge (senyal sinusoïdal de 230 V AC ), s'ha de tenir present que aquesta senyal pot presentar una sèrie de disturbis elèctrics tals com :

#### Interrupcions del servei (Blackout)



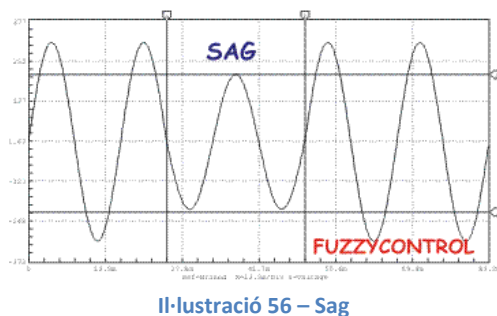
Succeeix quan la companyia elèctrica, per motius de manteniment decideix talle el subministrament d'energia elèctrica durant un període de temps curt.

#### Microtalls en el voltatge



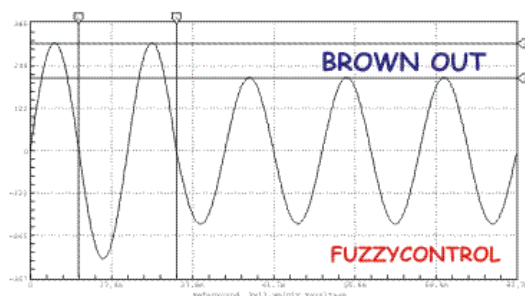
Són caigudes molt breus (en el rang de microsegons o pocs mil·lisegons) del subministrament d'energia elèctrica.

#### Baix voltatge momentani (Sag)



És un decrement en el nivell de voltatge de entre un 10 % i un 90 %, de una duració de entre un cicle i un i mig.

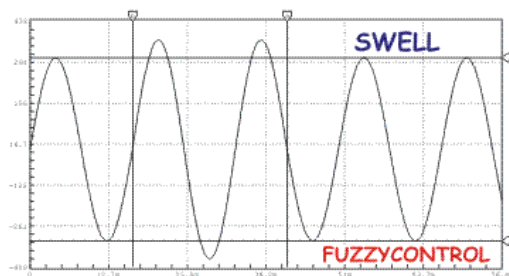
### Baix voltatge permanent (Brownouts)



Il·lustració 57 – Brownouts

És un decrement en el nivell de voltatge per sota del 90 %, de una duració de més d'un minut.

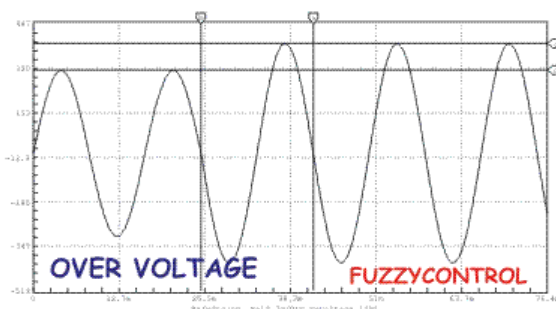
### Sobrevoltatge momentani (Swell)



Il·lustració 58 – Swell

És un augment en el nivell de voltatge per sobre del 110 % del valor nominal, de una duració de un cicle i un mig.

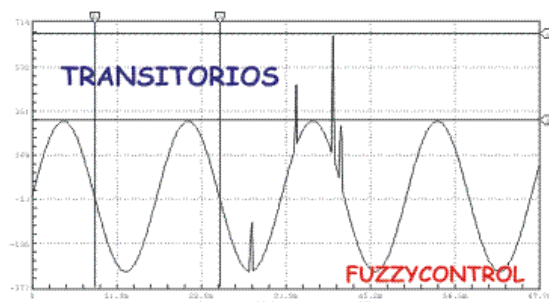
### Sobrevoltatge permanent (Over voltage)



Il·lustració 59 – Over voltage

És un increment en el nivell de voltatge per sota del 110 %, de una duració de més d'un minut.

### Sobrevoltatges transitoris (Transients)

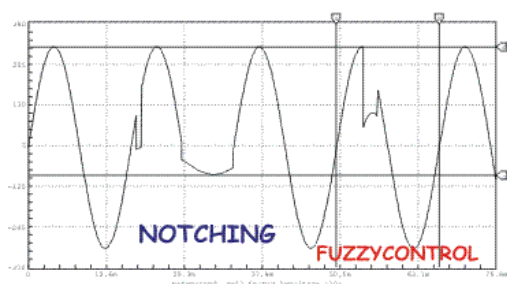


Il·lustració 60 – Transients

Són augments de voltatge espontanis de molt poca duració i d'amplitud variada.



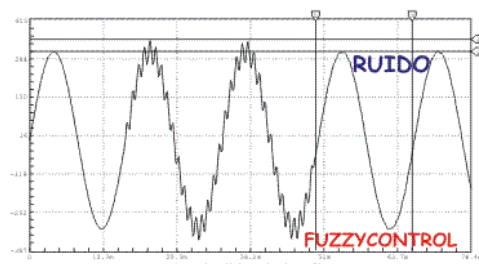
## Notching



Il·lustració 61 – Notching

És una pertorbació de polaritat oposada a la forma d'ona normal, d'una duració de mig cicle aproximadament.

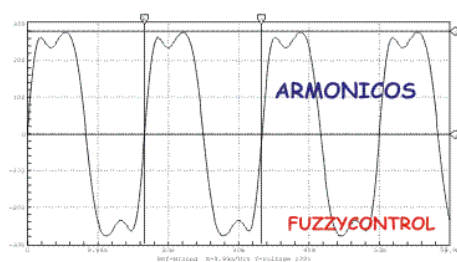
## Soroll elèctric



Il·lustració 62 – Soroll elèctric

És una distorsió d'alta freqüència en la forma d'ona del voltatge.

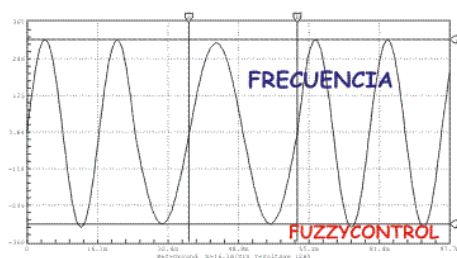
## Harmònics



Il·lustració 63 – Harmònics

Són una distorsió o deformació periòdica de la d'ona del voltatge.

## Canvi de freqüència del senyal

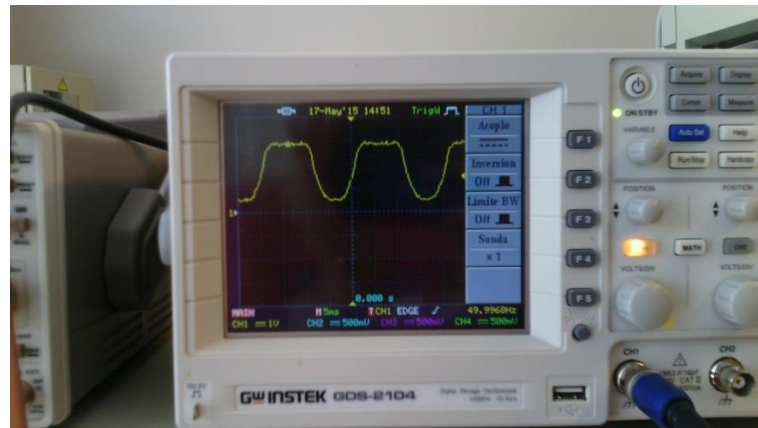


Il·lustració 64 – Canvi de la freqüència del senyal

Quan la freqüència del senyal canvia a valors allunyats de 50 Hz, pot provocar que els aparells no funcionin correctament.

Per comprova que el senyal d'entrada de 230 AC, no pateixi cap de les distorsions elèctriques abans mencionades, el que s'ha de fer és mirar amb l'oscil·loscopi a la sortida del divisor de tensió que transforma l'escala de 0 – 400 V a l'escala 0 – 0,15 V. Per tal de que el monitor d'energia pugui detectar correctament el senyal de 230 V AC, la forma d'ona que es visualitzaria hauria de ser molt semblant a la de una sinusoïdal ideal, a més hauria de tenir una freqüència de 50 HZ i l'amplitud màxima que hauria de tenir seria de 0,1218 V, que corresponen a 325 volts d'amplitud en l'escala de 0 – 400V.

- El rang de treball del optoacoblador amb el que s'ha treballat no és perfectament lineal, el que comporta que la senyal de tensió de sortida del optoacoblador no es serà una senyal perfectament sinusoidal. A continuació es mostra una il·lustració que mostra la tensió de sortida del optoacoblador, treballant amb una senyal d'entrada sinusoidal de 230 V AC, mitjançant un oscil·loscopi digital.



Il·lustració 65 – Senyal de sortida del optoacoblador

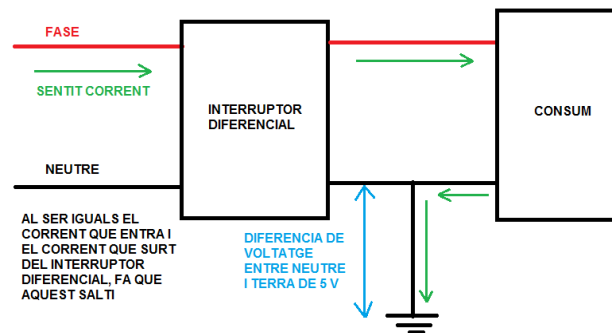
- Per connectar el circuit analògic a la placa ToolStick i conseqüentment connectar-ho al ordinador, s'han d'utilitzar 3 adaptadors (un pel monitor, un per a la font d'alimentació del ordinador i un per la regleta) per eliminar la presa a terra, degut a que si no es posen el interruptor diferencial salta i es talla el subministrament d'energia elèctrica. Llavors perquè salta el interruptor diferencial?

Normalment en les instal·lacions dels habitatges hi ha una diferencia de voltatge entre el neutre i la presa a terra d'uns 5 volts. En el circuit analògic del prototip del monitor, el neutre va connectat a la massa de la placa i conseqüentment a la massa del ordinador.

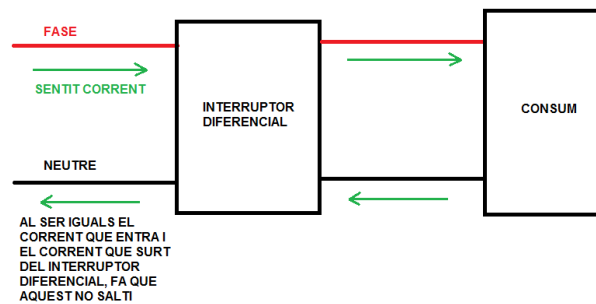
En l'ordinador internament, la massa d'aquest va connectada a la presa de terra, per tant de manera indirecte s'estaria connectant el neutre amb la presa a terra, és a dir, que el corrent que hauria de retornar el neutre cap al interruptor diferencial, s'aniria cap a la presa de terra i per tant el interruptor diferencial veuria que el corrent d'entrada no coincideix amb el corrent de sortida, fent que aquest salti i interrompi el corrent.



Il·lustració 66 – Adaptador per eliminar la presa a terra



Il·lustració 67 – Sentit del corrent si no es posen els adaptadors



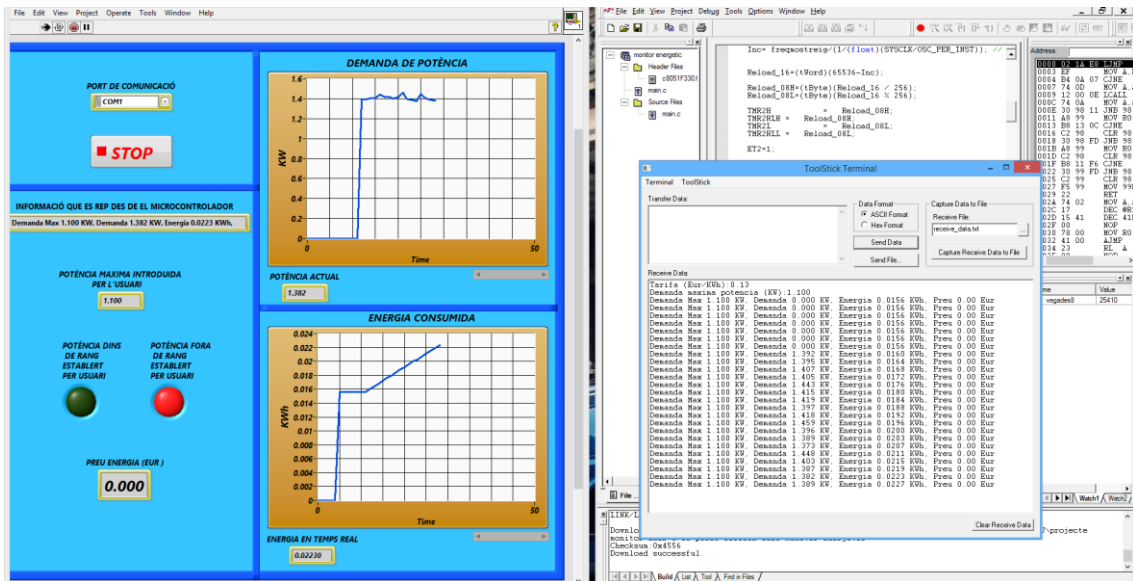
Il·lustració 68 – Sentit del corrent si es posen els adaptadors

El desavantatge que hi ha al posar aquests adaptadors és que s'estaria “enganyant” al interruptor diferencial perquè no salti i per tant no protegiria a persones en cas de descarrega. Per solucionar aquest problema i no haver de eliminar la presa a terra, el que es faria és no utilitzar la placa ToolStick F330 DC i utilitzar un microcontrolador PIC juntament amb una placa programadora de PIC's, ja que d'aquesta manera el neutre no es connectaria a la presa a terra.

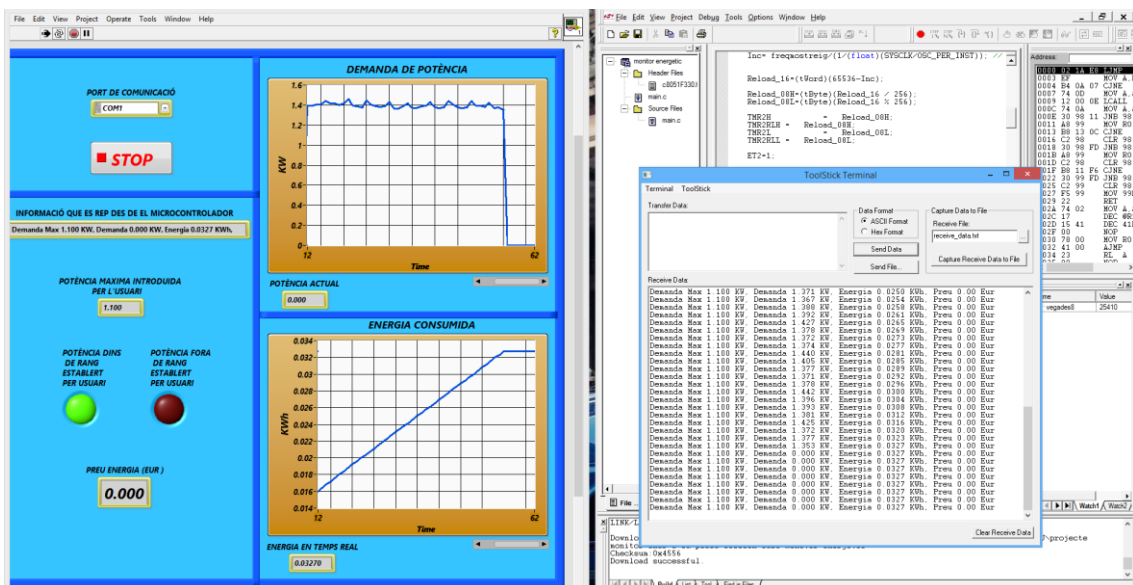
- La última consideració que s'ha de tenir en compte és que per mesurar el corrent que passa per la fase (que és el mateix que circula pel neutre ) es necessari tallar el cable de la fase de la regleta, per poder-lo passar pel mig del sensor.



# Consum de la paellera elèctrica

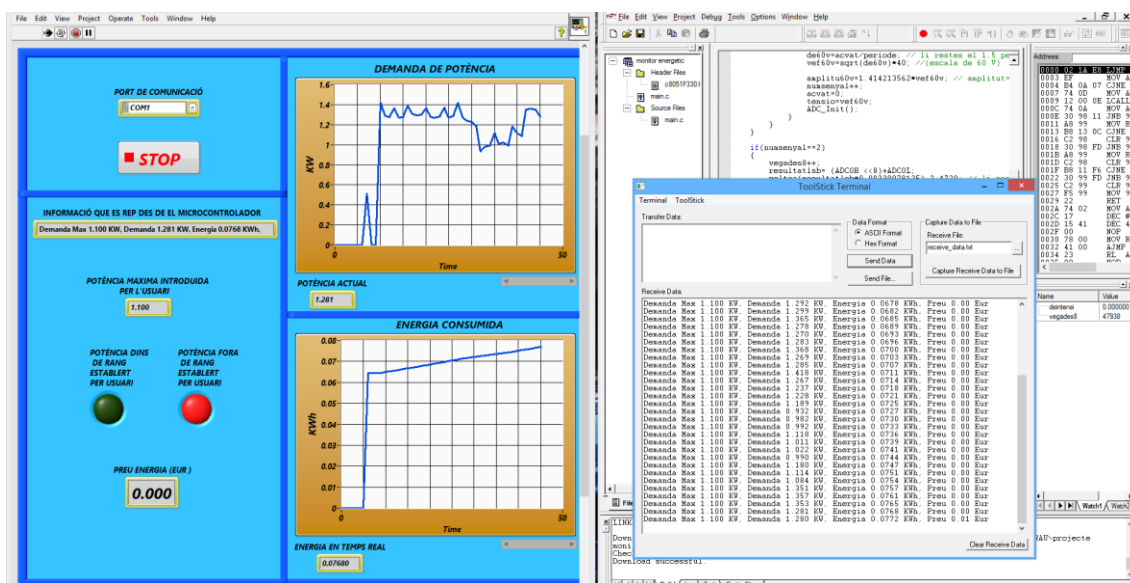


II-l·lustració 71 – Consum paellera elèctrica durant els primers 30 segons

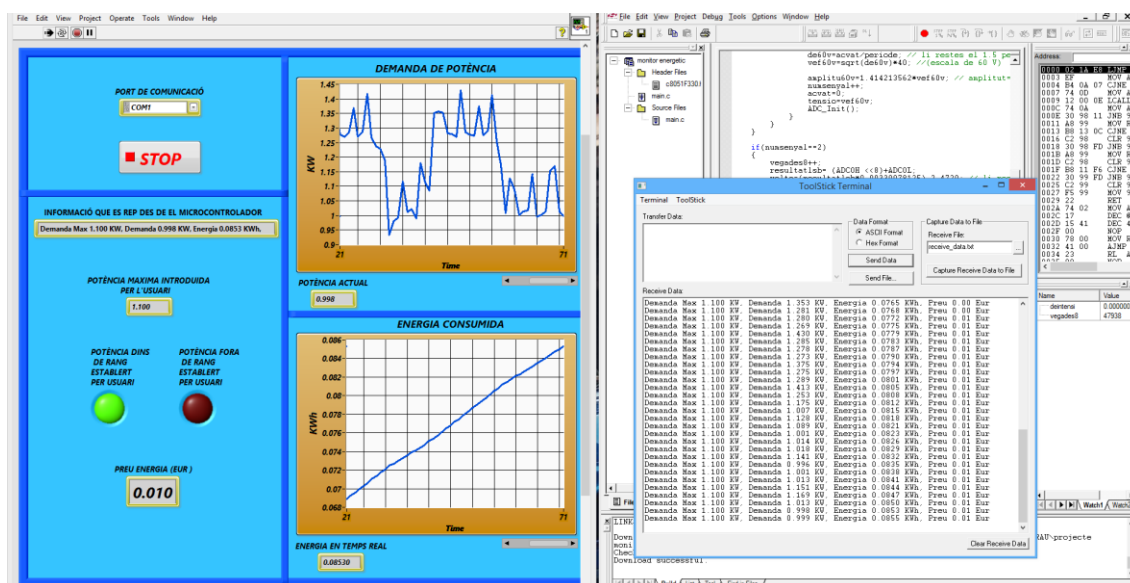


II-l·lustració 72 – Consum paellera elèctrica des de els 12 fins als 62 segons

## Consum del assecador de cabells



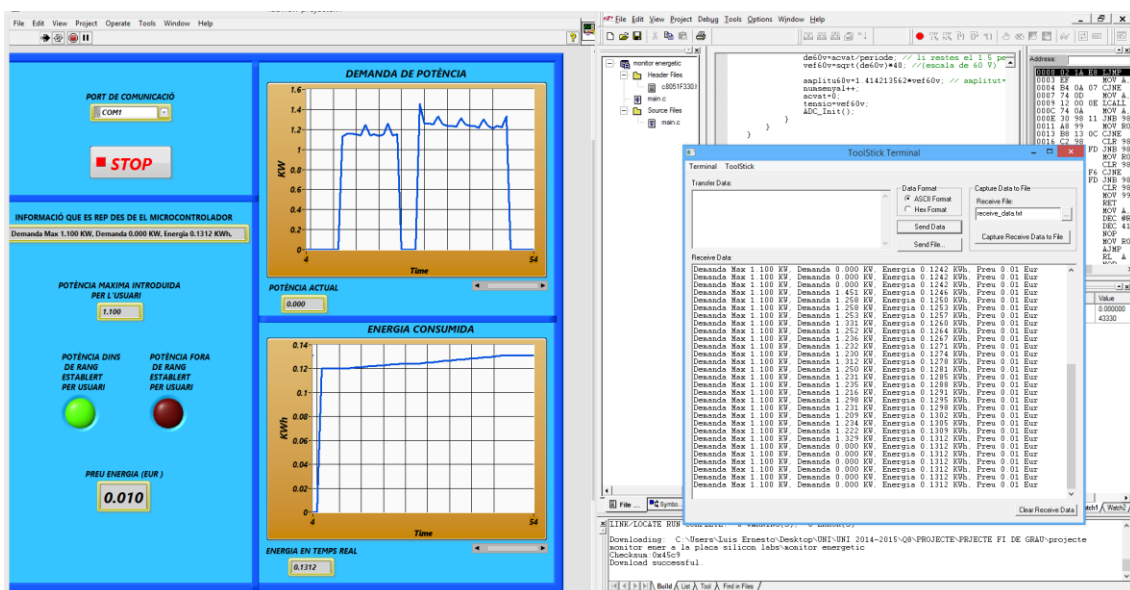
Il·lustració 73 – Consum assecador de cabells durant els primers 45 segons



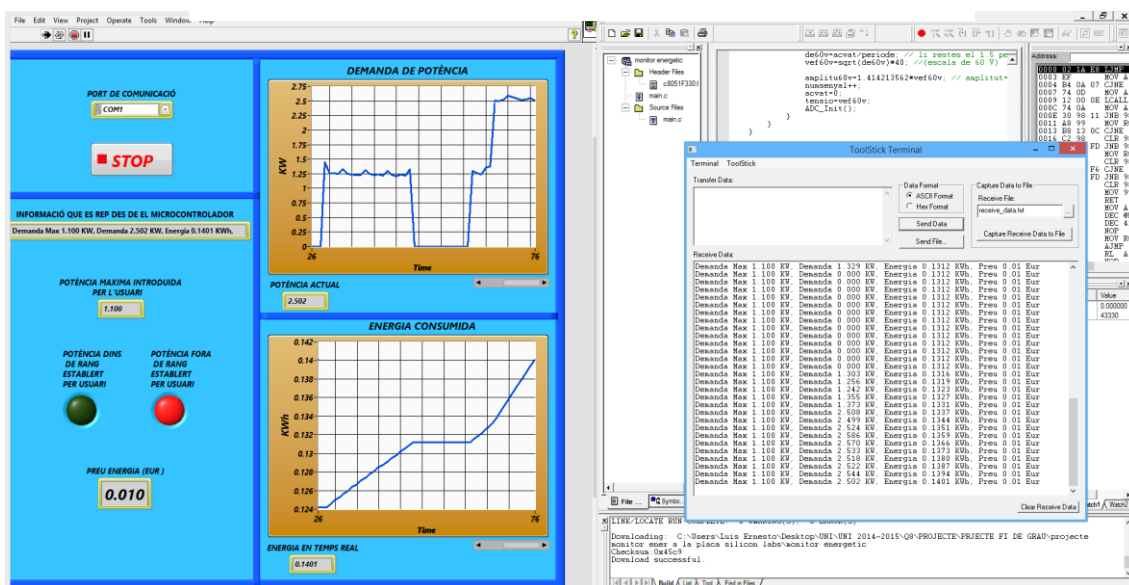
Il·lustració 74 – Consum assecador de cabells des de els 21 fins als 71 segons



## Consum del assecador de cabells, de l'aspiradora i de la suma dels 2 aparells



Il·lustració 75 – Consum assecador de cabells des de 10 fins 25 segons i aspiradora des de 27 fins a 50 segons



Il·lustració 76 – Consum aspiradora des de 27 fins a 50 segons i conjunt aspiradora-assecador de cabells des de 60 fins a 76 segons

Com es pot observar a les il·lustracions anteriors, les mesures de demanda de potència actual tenen un error de entre 20 i 120 wats aproximadament. Aquest error ve donat per una sèrie de limitacions, les quals que han estat condicionades per la resolució del sensor de corrent, el zero de referència de la sortida del sensor, de l'ajust dels potenciómetres i del rang de tensió de sortida quasi lineal del optocobrador.

## 7 Pressupost

Aquest apartat està destinat a enunciar els costos de cada element que componen el prototip del monitor d'energia elèctrica. Cal comentar que només s'enunciaren els costos dels materials utilitzats, quedant excloents els costos que li correspondrien a l'enginyer per haver programat el microcontrolador Intel 8051F330, així com també la programació de l'adquisició de dades amb LabVIEW. També quedarien excloents els costos corresponents al disseny del circuit analògic, al disseny i fabricació de la placa de circuit imprès, al muntatge i a la soldadura dels components. Cal destacar també que es farà un pressupost per al prototip del monitor d'energia d'aquest projecte que es basa en una comunicació amb fils, i un altre pressupost per un monitor d'energia elèctrica que es basi en una comunicació sense fils.

### Monitor d'energia elèctrica amb fils

Descripció	Quantitat	Preu	Total
ToolStick F330 DC amb adaptador USB	1	29,82	29,82
Cable DB9 M/F, RS232 1.8m	1	3,59	3,59
Adaptador DB9 M/F	1	3,45	3,45
Condensador electrolític 1 $\mu$ F 63V	5	0,08	0,40
Condensador ceràmic 1 $\mu$ F 50V	2	0,08	0,16
Terminals de 2 connexions	5	0,32	1,6
Tira pinheads	1	1,09	1,09
Connector pila 9V	2	0,49	0,98
Porta piles AA	1	0,68	0,68
Potenciòmetre ajustable	7	0,23	1,61
TL082 (2 OPAMS)	2	0,82	1,64
Optoacoblador TLP621	1	1,00	1,00
MAX232	1	3,18	3,18
Resistència 1/2w	1	0,04	0,04
Resistència 1/4w	15	0,02	0,3
Sensor de corrent HONEYWELL +/- 100A	1	18,40	18,40
Regleta de 3 endolls	1	3,65	3,65
Cablejat	1	2,50	2,50
Pila 9V SONY	2	3,10	6,20
Pila AA 1,5V	2	1,02	2,04
<b>TOTAL</b>			<b>82,33 €</b>



### Monitor d'energia elèctrica sense fils

Descripció	Quantitat	Preu	Total
ToolStick F330 DC amb adaptador USB	1	29,82	29,82
Condensador electrolític 1 $\mu$ F 63V	5	0,08	0,40
Condensador ceràmic 1 $\mu$ F 50V	2	0,08	0,16
Terminals de 2 connexions	5	0,32	1,6
Tira pinheads	1	1,09	1,09
Connector pila 9V	2	0,49	0,98
Porta piles AA	1	0,68	0,68
Potenciòmetre ajustable	7	0,23	1,61
TL082 (2 OPAMS)	2	0,82	1,64
Optoacobrador TLP621	1	1,00	1,00
MAX232	1	3,18	3,18
Resistència 1/2w	1	0,04	0,04
Resistència 1/4w	15	0,02	0,3
Sensor de corrent HONEYWELL +/- 100A	1	18,40	18,40
Regleta de 3 endolls	1	3,65	3,65
Cablejat	1	2,50	2,50
Pila 9V SONY	2	3,10	6,20
Pila AA 1,5V	2	1,02	2,04
Mòdul Bluetooth Slave/ Master	2	19,98	39,96
<b>TOTAL</b>			<b>115,25 €</b>

## 8 Estudi de viabilitat

En aquest apartat es vol fer un petit estudi de viabilitat del prototip del monitor d'energia elèctrica observant quins serien els principals competidors i a quins clients aniria destinat el producte.

### 8.1 A quins clients aniria destinat el producte?

El monitor d'energia elèctrica aniria destinat als següents clients:

- A totes aquelles persones que volguessin tenir un control exhaustiu del consum d'energia elèctrica de les instal·lacions dels seus habitatges, per poder reduir el cost de la factura de la llum.
- Persones que comprin electrodomèstics energèticament eficients i vulguin comprovar que el consum d'aquests concorda amb el consum anunciat pel fabricant.
- A petites empreses que tinguin màquines engegades durant moltes hores, i puguin detectar, per exemple si el consum de potència de la màquina sobrepassa un valor recomanat pel fabricant.

### 8.2 Com es donaria a conèixer el producte?

Per donar a conèixer el producte s'estudiaria la possibilitat d'assistir a fires d'energies renovables, per realitzar demostracions ja fos en les pròpies fires o en un local propi. Així mateix s'utilitzaria una de les eines més potents en el que a publicitat es refereix, Internet, fent anuncis gratuïts en varies pàgines. Així com també crear una pàgina web a on el client pugues accedir i veure demostracions del funcionament del producte.

### 8.3 Principals competidors

La llista dels principals competidors a Espanya és la següent:

- OWL
- Efergy
- HiperElectron
- Current Cost
- Mirubee

## 8.4 Es viable el producte?

El monitor d'energia elèctrica d'aquest projecte seria viable a partir de la reducció de costos en la fabricació dels prototips. Aquesta reducció de costos dependria de les relacions amb els proveïdors, això doncs si s'augmentés considerablement la comanda dels materials necessaris, els proveïdors, per conseqüent, podrien reduir el cost unitari dels materials.

Un altre factor que podria abaratir dràsticament els costos de producció consistirà en no utilitzar la placa toolstick F330 DC i comprar un microcontrolador que tingui un mínim de 3 ports d'entrada i que sigui barat.

Es podria agafa per exemple un microcontrolador anomenats PIC de l'empresa Microchip, que pot valdre al voltant de 10 euros, i fent el circuit corresponent per alimentar-lo costaria uns poc euros de més. A part es necessitaria una placa programadora, per carregar els codis als microcontroladors, que costat al voltant de 40 euros. Si s'escull aquesta opció el cost de fabricació es reduiria molt, ja que si es compra un placa ToolStick, aquesta només serviria per un microcontrolador (ja que la placa programadora del microcontrolador Intel 8051F330 ve incorporada al adaptador USB), mentre que si es compra un placa programadora per PIC's aquesta pot servir per programar més d'un microcontrolador PIC.

A continuació s'exposarà la reducció del cost que suposaria si enlloc de utilitzar la placa ToolStick, s'utilitzessin microcontroladors PIC.

L'exemple es posarà per la fabricació de 3 monitors d'energia elèctrica.

### Utilitzant les plaques ToolStick F330

Descripció	Quantitat	Preu	Total
ToolStick F330 DC amb adaptador USB	3	29,82	89,46

### Utilitzant microcontroladors PIC i una placa programadora

Descripció	Quantitat	Preu	Total
Placa programadora de PIC's	1	40,01	40,01
MICROCHIP PIC16F877-20/P MCU, 8BIT, PIC16, 20MHZ, DIP-40	3	7,64	22,92

**TOTAL** 62,93 €

Si es fabriquessin 3 monitors d'energia elèctrica i si s'utilitzen els PIC's es faria una reducció en el cost de fabricació de 89,46 – 62,93 = 26,53 €

## 9 Conclusions

En el present projecte s'ha estudiat, desenvolupat i implementat el prototip d'un monitor d'energia elèctrica per poder mesurar els consums d'energia elèctrica de diferents aparells, així com també el preu que comporten aquests consums. Els aparells que s'han utilitzat per realitzar les proves amb el prototip del mononitor d'energia elèctrica són : un assecador de cabells que té una potència mínima de 0,6 KW i una potència màxima de 1,2 KW, una paellera elèctrica amb una potència fixa de 1,5 W i una aspiradora amb una potència mínima de 1 KW i una potència màxima de 1,4 KW.

Aquests monitor d'energia es basa en un circuit analògic, muntat en una placa de circuit imprès, que s'encarrega d'adaptar el senyal de tensió d'entrada i el senyal de sortida del sensor de corrent per posteriorment introduir aquests senyals al microcontrolador Intel 8051F330, que amb la ajuda del convertidor Analògic – Digital li permetrà fer els càlculs per finalment enviar-los a una central, que en aquest cas es tracta d'un ordinador.

Com es pot comprovar, gracies als resultats obtinguts i exposats en aquesta memòria, s'han vist complerts els objectius proposats en un principi per desenvolupa un sistema que fos capaç de calcula el consum elèctrics d'una sèrie d'aparells i enviar els resultats a un ordinador perquè posteriorment aquest pogués processar-los.

El primer objectiu que s'ha complert ha consistit en entendre tots aquells conceptes racionats amb el funcionament d'un monitor d'energia. És a dir, per exemple entendre com es podien adaptar senyals d'entrada de magnituds de l'ordre de 400 volts a tensions que pugues llegir i interpretar un microcontrolador. Entendre també que el sensor de corrent funcionava gracies al fenomen físic anomenat efecte Hall que permet detectar el corrent que circula per un conductor a traves de la interacció del camp elèctric que genera dit conductor i un camp magnètic generat exteriorment sense haver de posar cap element en sèrie com seria el cas en el que es vulgues saber el corrent amb un multímetre. Entendre també que la comunicació entre el microcontrolador i l'ordinador es podia fer mitjançant la comunicació RS232 amb fils, però que també es podrien comunicar sense fils ajuntant la comunicació RS232 i la comunicació Bluetooth.

El segon objectiu que s'ha complert ha consistit en saber diferenciar i dividir correctament les parts principals que constitueixen un monitor d'energia l'elèctrica. Dites es van dividir en :

- Placa ToolStick, que incorpora el microcontrolador amb el qual es va treballar, així com també un adaptador USB que permet reprogramar el microcontrolador tantes vegades com es vulgui de manera molt senzilla. També incorporar un software que permet emular un display i un teclat que permeten l'enviament o la recepció d'informació des de o cap a l'exterior.

- Sensor de corrent que detectar el corrent que passa per la fase i envia un senyal de tensió proporcional al corrent que ha detectat.
- Comunicació entre l'ordinador i el microcontrolador, que permet l'enviament continu de resultats des de el microcontrolador cap a l'ordinador, perquè aquest últim sigui capaç de presentar dits resultats a l'usuari.
- Circuit analògic que facilita l'adaptació dels senyals d'entrada i de sortida del sensor de corrent perquè el microcontrolador pugui llegir aquestes senyals i pugui posteriorment realitzar càlculs i obtenir uns resultats.

El tercer objectiu que s'ha complert ha consistit en ampliar el coneixements de programació. S'han pogut ampliar tant en la programació del microcontrolador com en la programació de la recepció dels resultats. Aquesta ampliació de coneixements no hagués estat possible sense l'enteniment d'una sèrie de conceptes bàsics en programació, dita base s'ha adquirit gràcies a l'ensenyament proporcionat per part de la universitat.

El quart objectiu que s'ha complert ha consistit en poder fer la realització del prototip del monitor d'energia elèctrica amb una baixa inversió. Cal destacar per això que molts dels materials utilitzats en el prototip han estat proporcionats per la universitat.

El cinquè i últim objectiu complert ha consistit en assegurar un bon funcionament del prototip del monitor d'energia elèctrica dins d'uns límits acceptables. Aquests límits han estat condicionats per la resolució del sensor de corrent, el zero de referència de la sortida del sensor, de l'ajust dels potenciómetres i del rang de tensió de sortida quasi lineal del optoacoblador. Tots aquests factors han condicionat la mesura final dels consums dels aparells, que ha comportat que a cada mesura hi hagi un error entre 50 i 120 wats aproximadament.

Pel que fa referent a la valoració del projecte, aquesta es pot considerar molt positiva, ja que s'han ampliat els coneixements que d'una manera o altre són complementaris als coneixements adquirits a la universitat. A més la fabricació del prototip del monitor d'energia s'ha completat satisfactòriament i el seu funcionament és prou bo tenim en compte les limitacions abans esmentades.

## 10 Bibliografia

### Pàgines web

[1] ¡Ojo al Vatio! Aparatos para vigilar la electricidad que consumimos, Consultat: [en la xarxa]. [Accés 3 de Març 2015]. Disponible a:

<http://www.ennaranja.com/para-ahorradores/ojo-al-vatio-aparatos-para-vigilar-la-electricidad-que-consumimos/>

[2] Uso de un monitor de Energía para Controlar sus Costos de Electricidad, Consultat: [en la xarxa]. [Accés 3 de Març 2015]. Disponible a:

<http://www.ext.colostate.edu/pubs/spanish/10628.html>

[3] Controla tu consumo eléctrico con Efergy, Consultat: [en la xarxa]. [Accés 3 de Març 2015]. Disponible a:

<http://www.xatakahome.com/iluminacion-y-energia/controla-tu-consumo-electrico-con-efergy>

[4] ACS712 current sensor , Consultat: [en la xarxa]. [Accés 10 de Març 2015]. Disponible a:

<http://embedded-lab.com/blog/?p=4469>

[5] Energía eléctrica y medio ambiente , Consultat: [en la xarxa]. [Accés 10 de Març 2015]. Disponible a:

[http://www.endesaeduca.com/Endesa\\_educa/recursos-interactivos/el-uso-de-la-electricidad/xxv.-la-energia-electrica-y-el-medio-ambiente](http://www.endesaeduca.com/Endesa_educa/recursos-interactivos/el-uso-de-la-electricidad/xxv.-la-energia-electrica-y-el-medio-ambiente)

[6] OPTOACOPLADOR, Consultat: [en la xarxa]. [Accés 11 de Març 2015]. Disponible a:

<http://www.areatecnologia.com/electronica/optoacoplador.html>

[7] Valor eficaz , Consultat: [en la xarxa]. [Accés 11 de Març 2015]. Disponible a:

[https://es.wikipedia.org/wiki/Valor\\_eficaz](https://es.wikipedia.org/wiki/Valor_eficaz)

[8] Generación de energía eléctrica , Consultat: [en la xarxa]. [Accés 11 de Març 2015]. Disponible a:

[https://es.wikipedia.org/wiki/Generaci%C3%B3n\\_de\\_energ%C3%ADa\\_el%C3%A9ctrica](https://es.wikipedia.org/wiki/Generaci%C3%B3n_de_energ%C3%ADa_el%C3%A9ctrica)

[9] Dígitos binarios , Consultat: [en la xarxa]. [Accés 14 de Març 2015]. Disponible a:

<https://electronicados.wordpress.com/2012/11/22/digitos-binarios-niveles-logicos-y-formas-de-onda-digitales/>

[10] La comunicación serie , Consultat: [en la xarxa]. [Accés 15 de Març 2015]. Disponible a:

<http://perso.wanadoo.es/pictob/comserie.htm>

[11] Conversión a ASCII , Consultat: [en la xarxa]. [Accés 21 de Març 2015]. Disponible a:

<http://perso.wanadoo.es/pictob/tecprg.htm>

[12] HONEYWELL S&C CSLT6B100 CURRENT SENSOR , Consultat: [en la xarxa]. [Accés 21 de Març 2014]. Disponible a:

<http://es.farnell.com/honeywell-s-c/cslt6b100/current-sensor--100a-4-5vdc-to/dp/2070563>

[13] Comunicación serial utilizando LabVIEW , Consultat: [en la xarxa]. [Accés 30 de Març 2015]. Disponible a:

<http://www.ni.com/white-paper/7907/es/>

[14] Bluetooth , Consultat: [en la xarxa]. [Accés 4 de Abril 2015]. Disponible a:

<https://es.wikipedia.org/wiki/Bluetooth>

[15] Serial port Bluetooth with expansionBoard , Consultat: [en la xarxa]. [Accés 4 de Abril 2015]. Disponible a:

<http://emartee.com/product/41934/Serial-Port-Bluetooth-With-ExpansionBoard>

[16] Suma de Riemann , Consultat: [en la xarxa]. [Accés 9 de Abril 2015]. Disponible a:

<http://www.zweigmedia.com/MundoReal/integral/numint.html>

[17] Tarifas Luz Endesa , Consultat: [en la xarxa]. [Accés 14 de Abril 2015]. Disponible a:

<http://www.rankia.com/blog/luz-y-gas/2688319-mejores-tarifas-luz-gas-marzo-2015>

[18] Calidad de energía eléctrica , Consultat: [en la xarxa]. [Accés 4 de Abril 2015]. Disponible a:

<http://www.fuzzycontrolsac.com/articulos/articulo1.htm>

[19] TOOLSTICK , Consultat: [en la xarxa]. [Accés 30 de Abril 2015]. Disponible a:

<http://www.mouser.es/ProductDetail/Silicon-Labs/TOOLSTICKSK/?qs=AG1tZYOK7s69qVoZbA6BRw%3D%3D>

[20] MICROCHIP PIC16F877 , Consultat: [en la xarxa]. [Accés 10 de Maig 2015]. Disponible a:

<http://es.farnell.com/microchip/pic16f877-20-p/mcu-8bit-pic16-20mhz-dip-40/dp/9761373?mckv=CkkWKZf5|pcrid|productlistings&CMP=KNC-GES-FES-GEN-LISTINGS>

[21] Targeta de programación PIC , Consultat: [en la xarxa]. [Accés 10 de Maig 2015]. Disponible a:

[http://jr-international.fr/kit-para-montar-tarjeta-programacion\\_K8076\\_itm\\_spanish\\_EUR\\_.html?gclid=CL6mkMyaysYCFE\\_HtAodxFoLpA](http://jr-international.fr/kit-para-montar-tarjeta-programacion_K8076_itm_spanish_EUR_.html?gclid=CL6mkMyaysYCFE_HtAodxFoLpA)

## **Arxius PDF**

[1] Monitores de energia , Consultat: [en la xarxa]. [Accés 3 de Març 2015]. Disponible a:

<http://www.comei.es/sites/default/files/pdf/monitor-energia.pdf>

[2] TOOLSTICK C8051F330 , Consultat: [en la xarxa]. [Accés 12 de Març 2015]. Disponible a:

<https://www.silabs.com/Support%20Documents/TechnicalDocs/ToolStick-F330-DC-UG.pdf>

[3] HONEYWELL SENSOR CURRENT , Consultat: [en la xarxa]. [Accés 16 de Març 2015]. Disponible a:

<http://www.farnell.com/datasheets/1676939.pdf>

[4] TOSHIBA TLP621 , Consultat: [en la xarxa]. [Accés 17 de Març 2015]. Disponible a:

<http://www.glyn.de/data/glyn/media/doc/TLP621.pdf>



## Vídeos

[1] Hall effect based current sensors , Consultat: [en la xarxa]. [Accés 18 de Març 2015]. Disponible a:

[https://www.youtube.com/watch?v=C\\_FgYMmFGQY](https://www.youtube.com/watch?v=C_FgYMmFGQY)

[2] Serial communication using Labview , Consultat: [en la xarxa]. [Accés 30 de Març 2015]. Disponible a:

<https://www.youtube.com/watch?v=RM8a66g2eGo>